EMBEDDIA

Cross-Lingual Embeddings for Less-Represented Languages in European News Media

Research and Innovation Action
Call: H2020-ICT-2018-1
Call topic: ICT-29-2018 A multilingual Next generation Internet
Project start: 1 January 2019 Project duration: 36 months

D2.7: Final evaluation report on advanced cross-lingual NLP technology (T2.4)

Executive summary
This final report describes the resources collection, benchmarks and evaluation gathered for WP2 'Advanced NLP Technologies for Less-Resourced Languages'. Specifically, it focuses on WP2's tasks related to document enrichment, T2.1 and T2.2. For each of these tasks, we present a list of selected datasets available in the literature, and present novel datasets, developed during the project. Next, we present new enrichment methods which have been developed since M24 deliverables. Finally, we evaluate the developed enrichment methods on the gathered datasets and compare their performance against tools from the state of the art.

Partner in charge: ULR

Project co-funded by the European Commission within Horizon 2020 Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	–
RE	Restricted to a group specified by the Consortium (including the Commission Services)	–
CO	Confidential, only for members of the Consortium (including the Commission Services)	–

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825153

Deliverable Information

Document administrative information	
Project acronym:	**EMBEDDIA**
Project number:	**825153**
Deliverable number:	**D2.7**
Deliverable full title:	**Final evaluation report on advanced cross-lingual NLP technology**
Deliverable short title:	**Final evaluation of cross-lingual NLP technology**
Document identifier:	**EMBEDDIA-D27-FinalEvaluationOfCrosslingualNLPTechnology-T24-submitted**
Lead partner short name:	**ULR**
Report version:	**submitted**
Report submission date:	**30/06/2021**
Dissemination level:	**PU**
Nature:	**R = Report**
Lead author(s):	**Luis Adrián Cabrera-Diego (ULR), Emanuela Boros (ULR), Thi Hong Hanh Tran (ULR)**
Co-author(s):	**Elvys Linhares Pontes (ULR), Jose G. Moreno (ULR), Antoine Doucet (ULR), Senja Pollak (JSI), Matej Martinc (JSI), Andraž Repar (JSI)**
Status:	**__ draft, __ final, x submitted**

The EMBEDDIA Consortium partner responsible for this deliverable has addressed all comments received. Changes to this document are detailed in the change log table below.

Change log

Date	Version number	Author/Editor	Summary of changes made
19/01/2021	v0.1	Elvys Linhares Pontes (ULR)	Structure proposal.
20/04/2021	v0.2	Luis Adrián Cabrera-Diego (ULR)	Structure modification.
21/05/2021	v0.3	Luis Adrián Cabrera-Diego and Emanuela Boros (ULR)	Sections regarding Named Entity Recognition and Linking, and Event Detection.
25/05/2021	v0.4	Thi Hong Hanh Tran (ULR) and Matej Martinc (JSI)	Sections on keyword and term extraction.
01/06/2021	v0.5	Andraž Repar (JSI)	Section on term alignment.
03/06/2021	v0.6	Senja Pollak (JSI)	Revision of sections on keywords and term extraction and alignment.
04/06/2021	v0.7	Luis Adrián Cabrera-Diego (ULR)	Normalisation of tables and minor fixes.
14/06/2021	v0.8	Hannu Toivonen (UH)	Internal review.
20/06/2021	v0.9	Ravi Shekhar (QMUL)	Internal review.
25/06/2021	v1.0	All authors	Internal review changes.
28/06/2021	prefinal	Nada Lavrač (JSI)	Quality control finalized.
28/06/2021	final	Luis Adrián Cabrera-Diego (ULR)	Report finalized.
30/06/2020	submitted	Tina Anžič (JSI)	Report submitted.

Table of Contents

List of abbreviations

ACE 2005	Automatic Content Extraction 2005
BSNLP	Balto Slavic Natural Language Processing
BERT	Bidirectional Encoder Representations from Transformers
DAnIEL	Data Analysis for Information Extraction in any Language
ED	Event Detection
EE	Event Extraction
F1	F1-score
JSI	Jožef Stefan Institute
NEL	Named Entity Linking
NER	Named Entity Recognition
NLP	Natural Language Processing
P@k	Precision at k
R@k	Recall at k
RoBERTa	Robustly optimized BERT approach
ULR	University of La Rochelle
WP	Work Package
XLM-RoBERTa	Cross-lingual RoBERTa

1 Introduction

In this deliverable, we present the results of WP2, *Advanced NLP Technologies for Less-Resourced Languages* related to the task T2.4 *Public resource gathering, benchmarking and evaluation*.

The global objective of WP2 is to enrich individual documents and their content with semantic information, through the use of embeddings. As well, it encloses the extraction of elements that could improve the access to information. Therefore, WP2 covers different aspects, from the recognition of named entities to the extraction of keywords, while passing through the linking of mentions and the detection of events.

Task T2.4 focuses on collecting different datasets and benchmarks, and uses them for the final evaluation of the document enrichment systems developed in WP2.This deliverable, entitled *Final evaluation report on advanced cross-lingual NLP technology*, presents the final report regarding the evaluation of tasks T2.1 and T2.2, which covered document enrichment technologies.

Since the first deliverable D2.1, the work performed in the scope of Task T2.4 resulted also in several publications and the following achievements:

- For Named Entity Recognition (NER), members of EMBEDDIA team co-organised the 3rd Slavic NER challenge (SlavNER) (Piskorski et al., 2021) (Appendix F) in the scope of the Balto-Slavic Natural language processing workshop at EACL 2021. This resulted in a new annotated dataset that covers Slavic languages, two of which are explored in EMBEDDIA, Russian and Slovene. The ULR team competed at SlavNER by proposing a NER system, described in (Cabrera-Diego, Moreno, & Doucet, 2021b) (Appendix A), which was trained using a Frustratingly Easy Domain Adaptation (Daumé III, 2007) and elements from our Multitask BERT, which were detailed in D2.5 and in (Cabrera-Diego, Moreno, & Doucet, 2021a).
- For Named Entity Linking (NEL), ULR were invited to submit a journal article regarding the linking of named entities (see submission in Appendix D), where we present an improved version of the models described in D2.5 as well as a more detailed results analysis. In this improved version, we explored weighted edit distance metrics for improving the linking of entities from documents processed with an OCR. Furthermore, we detail a post-processing filter which improves the accuracy of the predicted links by removing and reorganising the proposed linking candidates.
- For Event Detection (ED): The baseline has been defined as the performance obtained by the D2.2 NER system over the corpus SlavNER 2019, which covers events in Slavic languages. Our improvement is an increment of at least 39%. We also approached this task in the context of other datasets that are widely used in this field and we published several models that reached state-of-the-art results in (Boros, Moreno, & Doucet, 2021b), (Boros, Moreno, & Doucet, 2021a), (Mutuvi et al., 2020).
- For keyword extraction, we performed a systematic evaluation of neural keyword extraction systems (including our TNT-KID approach described in D2.3 (Martinc, Škrlj, & Pollak, 2021)) on novel benchmark datasets that we created, and proposed a method for improving the recall (Koloski, Pollak, Škrlj, & Martinc, 2021). We also perform novel cross-lingual keyword extraction experiments.
- For term extraction, we systematically evaluate different neural methods on term extraction benchmarks (Rigouts Terryn, Hoste, & Lefever, 2019).
- For term alignment, the two experiments from D2.6 were now finalized and turned into publications. In the first one (Repar & Shumakov, 2021), we applied our term alignment method (Repar, Martinc, & Pollak, 2019) to align ExM tags between Russian and Estonian, and in the second one (Repar, Martinc, Ulčar, & Pollak, 2021), we extended the term alignment method by using embeddings-based features.

In addition, several papers describing evaluation of our document enrichment methods that were submitted at the time of publication of D2.5 and D2.6, were now accepted or published. As these publica-

tions were listed in previous deliverables, we do not add them here as appendices, but the interested reader can refer to our paper (Cabrera-Diego et al., 2021a) for final evaluation of Multitask BERT, and to (Martinc et al., 2021) for the final version of TNT-KID keyword extraction. In addition, in collaboration with T4.3, we performed a sentiment analysis study on Slovene news, where our NER and NEL systems, (Boros, Hamdi, et al., 2020) and (Linhares Pontes, Cabrera-Diego, et al., 2020) respectively, were used. This work is published in (Valmarska, Cabrera-Diego, Linhares Pontes, & Pollak, 2021), and will be reported in the final deliverable of Task T4.3.

This report is organised as follows. In Section 2, we introduce the evaluation metrics used in this deliverable. Then, we present the final evaluation of Named Entity Recognition in Section 3, of Named Entity Linking in Section 4, of Event Detection in Section 5. As well, we present the final assessment for Multilingual Keyword Extraction Methods in Section 6, of Term Extraction in Section 7 and of Term and Keyword Alignment in Section 8. The conclusions and future work are set in Section 9. The associated outputs generated in deliverable are presented in Section 10.

2 Evaluation metrics

To asses the performance of keyword extraction models presented in this deliverable, we measure F1@k score, a harmonic mean between Precision@k and Recall@k. Precision is the ratio of the number of correct keywords returned by the system divided by the number of all keywords returned by the system, or more formally:

$$precision = \frac{|correct \quad returned \quad keywords@k|}{|returned \quad keywords|}$$

Recall@k is the ratio of the number of correct keywords returned by the system and ranked equal to or better than k divided by the number of correct ground truth keywords:

$$recall = \frac{|correct \quad returned \quad keywords@k|}{|correct \quad keywords|}$$

Finally, we formally define F1@k as a harmonic mean between Precision@k (P@k) and Recall@k (R@k):

$$F1 - score@k = 2 * \frac{P@k * R@k}{P@k + R@k}$$

Similarly, for Named entity Recognition (NER), Named Entity Linking (NEL) and Event Detection (ED), and term extraction, we evaluate our models using as well precision, recall and F-score.

Precision is the number of instances (entities, links, events or terms) correctly labeled by a system compared to the number of tagged instances returned by the same system.

$$precision = \frac{|correct \quad returned \quad instances|}{|returned \quad instances|}$$

Recall is the number of instances correctly labeled compared to the number of tagged instances in the reference.

$$recall = \frac{|correct \quad returned \quad instances|}{|correct \quad instances|}$$

Finally, the F1-score is defined as the harmonic mean of precision (P) and recall (R):

$$F1 - score = 2 * \frac{P * R}{P + R}$$

All the previously presented metrics, e.g., precision, precision@k, recall or recall@k, can be calculated using micro and macro averaging. The micro-averaging consists of summing the number of *correct*, *correct returned* and *returned* keywords/instances regardless if they come from multiple classes or documents. In contrast, macro-averaging calculates the metrics for each class/document beforehand and then the average is done. In most of the results presented in this deliverable we use micro-averaging, unless it is stated the opposite.

Finally, we present in Equation 1 the formula utilised for calculating the improvement with respect to the scores from the state of the art and those obtained by our best models.

$$Improvement = 100 \frac{Score_{new} - Score_{old}}{Score_{old}} \quad (1)$$

3 Final Evaluation of Named Entity Recognition (Task 2.1)

In this section, we present the evaluation of Named Entity Recognition (NER) systems developed for EMBEDDIA. We also introduce a list of datasets that are related to this NLP task.

3.1 Background

Named entity recognition (NER) is an NLP task which consists of tagging a word, or a group of them, with labels that make reference to semantic aspects such as locations, persons, and organisations (Luoma, Oinonen, Pyykönen, Laippala, & Pyysalo, 2020; Li, Sun, Han, & Li, 2020).

The first definition of "named entities" was proposed in 1996, at the 6th Message Understanding Conference (MUC). There, the organisers defined named entities as "the names of all the people, organisations, and geographic locations in a text" (Grishman & Sundheim, 1996). Since then, NER has become an NLP key task, either stand-alone or in conjunction with other tasks, such as automatic text summarisation, question-answering, and machine translation (Li et al., 2020). Furthermore, it has been applied in different domains, from newspapers, such as (Tjong Kim Sang, 2002; Tjong Kim Sang & De Meulder, 2003), to the biomedical domain like in (Li et al., 2016), while passing through more generic domains as in (Luoma et al., 2020).

Although initial works on NER focused on documents in English, they rapidly started to cover other languages, such as Spanish and German (Tjong Kim Sang, 2002), but also less-resourced languages like Swedish (Dalianis & Åström, 2001) and Lithuanian (Kapočiūtė & Raškinis, 2005).

With the creation of BERT (Devlin, Chang, Lee, & Toutanova, 2019), new NER systems have been proposed based on the fine tuning of language models. Related to the languages explored in EMBEDDIA, we can name: (Ulčar & Robnik-Šikonja, 2020) for Finnish, Slovene and Croatian; (Ljubešić & Lauc, 2021) for Croatian; Finnish (Virtanen et al., 2019); Swedish (Malmsten, Börjeson, & Haffenden, 2020); Russian (Arkhipov, Trofimova, Kuratov, & Sorokin, 2019; Kuratov & Arkhipov, 2019); Estonian (Tanvir, Kittask, & Sirts, 2020); Latvian (Znotiņš & Guntis Barzdiņš, 2020).

3.2 Available Resources

In Table 1, we present different datasets that contain annotations regarding Named Entity Recognition (NER) that can be found in the literature for the languages explored in EMBEDDIA.

Table 1: The collected datasets for the NER task and their properties: acronym, name, year of publication, availability, languages, and link to the corpus location.

Acronym	Name	Year	Public	Language	Location
FIN-CLARIN	Finnish News Corpus for Named Entity Recognition	2019	Yes	fi	link
WikiANN	Cross-lingual name tagging and linking for 282 languages	2017	Yes	282 languages (et, fi, hr, lt, lv, ru, sl, sv)	link
SlavNER 2017	1st Shared task on Slavic Named Entity Recognition	2017	Yes	cs, hr, pl, ru, sk, sl, uk	link
SlavNER 2019	2nd Shared task on Slavic Named Entity Recognition	2019	Yes	bg, cs, pl, ru	link
SlavNER 2021	3rd Shared task on Slavic Named Entity Recognition	2021	Yes	bg, cs, pl, ru, sl, uk	
SETimes.HR+	The SETimes.HR+ Croatian dependency treebank	2013	Yes	hr, sr	link link
GermEVAL2014	GermEval 2014 Named Entity Recognition Shared Task	2014	Yes	de	link
KaggleNER	Annotated Corpus for Named Entity Recognition	2017	Yes	en	link
EstNER	Estonian NER corpus	2013	Yes	et	link
Finer-data	A Finnish News Corpus for Named Entity Recognition	2014	Yes	fi	link
HR500k	Training corpus HR500k 1.0	2018	Yes	hr	link link
TildeNER	accurat-toolkit/TildeNER	2012	Yes	lt	link
LVTagger	PeterisP/ LVTagger/ NerTrainingData/	2013	Yes	lv	link
factRuEval-2016	factRuEval-2016 dialog-21.ru	2016	Yes	ru	link
SSJ500k	Training corpus SSJ500k 2.2	2019	Yes	sl	link link
Slovene news	Slovene news - slavko.zitnik	2011	Yes	sl	link
SwedishNER	Swedish manually annotated NER	2012	Yes	sv	link
Janes-Tag	CMC training corpus Janes-Tag 2.1	2019	Yes	sl	link
ReLDI-NormTag NER-hr	Croatian Twitter training corpus ReLDI-NormTagNER-hr 2.1	2019	Yes	hr	link
SIC	Stockholm Internet Corpus	2016	Yes	sv	link
Finnish NER	Broad-coverage Corpus for Finnish NER	2020	Yes	fi	
CNE5	Collection Named Entities 5	2016	Yes	ru	link

It should be indicated that the Jožef Stefan Institute (JSI) was co-organiser of the SlavNER 2021 challenge and participated in the annotation of dataset SlavNER 2021 (Piskorski et al., 2021). Specifically, JSI annotated the data that was made available for Slovene. The annotation consisted of marking named entities with their respective named entity linking. The paper is presented in Appendix F.

3.3 Results

In Deliverable D2.2, and published in (Moreno, Linhares Pontes, Coustaty, & Doucet, 2019), we proposed a NER system based on the work of (Reimers & Gurevych, 2019). This system consisted of using multiple types of embeddings, such as BERT and FastText, along with a BiLSTM to generate a sequence-to-sequence NER tagger.

For Deliverable D2.5, we explored different NER systems based on fine-tuned BERT models. The best performing NER systems, described in detail in (Cabrera-Diego et al., 2021a), was a Multitask BERT. Specifically, this NER systems consisted of training the NER system along with other tasks, such as prediction of masked tokens and entities boundaries, as well by marking specific tokens.

We present, in Table 2, the results obtained by the NER systems trained and tested over the dataset WikiANN. We can observe in Table 2 that Multitask BERT provide a better performance than the architecture proposed in Deliverable D2.2 based on a BiLSTM with multiple embeddings.

Table 2: F1-score obtained between the BiLSTM-based NER model (described in D2.2) and Multitask BERT (see D2.5) on the WikiANN dataset. The improvement percentage is calculated using Equation 1.

	F1-score				Improvement	
	BiLSTM		Multitask BERT		(%)	
Language	**Macro**	**Micro**	**Macro**	**Micro**	**Macro**	**Micro**
et	0.854	0.859	0.947	0.949	10.889	10.477
fi	0.850	0.855	0.939	0.941	10.470	10.058
hr	0.856	0.858	0.945	0.946	10.397	10.256
lt	0.842	0.843	0.921	0.922	9.382	9.371
lv	0.885	0.883	0.948	0.948	7.118	7.361
ru	0.844	0.841	0.917	0.915	8.649	8.799
sl	0.890	0.892	0.955	0.956	7.303	7.174
sv	0.903	0.909	0.956	0.959	5.869	5.500
Average	0.865	0.867	0.941	0.942	8.760	8.624

3.3.1 Participation in BSNLP - SlavNER 2021

The University of La Rochelle (ULR) participated in April 2021 at BSNLP - SlavNER 2021 (Piskorski et al., 2021), a challenge regarding the prediction of named entities in six Slavic languages: Bulgarian, Czech, Polish, Slovenian, Russian and Ukrainian.

Specifically, ULR participation (Cabrera-Diego et al., 2021b) (Appendix A) consisted on training multiple NER systems using different BERT models and a Frustratingly Easy Domain Adaptation (FEDA) (Daumé III, 2007; Kim, Stratos, & Sarikaya, 2016). The use of FEDA allowed ULR creating NER systems using multiple datasets regardless whether the tagset (e.g. Location, Event, Miscellaneous, Time) in the source and target domains matched. Furthermore, we applied some of the techniques explored in D2.5, i.e. uppercase words and predicting masked words, and which were described in (Cabrera-Diego et al., 2021a). We present, in Table 3, the results obtained by our models in terms of strict micro F1-Score at BSNLP - SlavNER 2021.[1] Overall, ULR's NER system was ranked on 2nd place, while it achieved the first place on languages such as Bulgarian and Russian.

[1] For a description of the strict micro F1-score see (Piskorski et al., 2021).

Table 3: ULR results at BSNLP - SlavNER 2021 based on strict micro F1-scores. The *Global* column is the strict micro F-score regarding all the test data.

	Covid-19							U.S. Elections							
Model	**Bg**	**Cs**	**Pl**	**Ru**	**Sl**	**Uk**	**All**	**Bg**	**Cs**	**Pl**	**Ru**	**Sl**	**Uk**	**All**	**Global**
Cyrillic-1	0.716	0.714	0.760	0.657	0.732	0.722	0.715	0.843	0.837	0.841	0.741	0.837	0.787	0.793	0.764
Cyrillic-2	0.720	0.730	0.783	0.642	0.744	**0.727**	0.721	**0.865**	0.857	0.849	0.746	0.858	**0.813**	0.807	0.775
Latin-1	0.730	0.765	0.791	0.662	0.752	0.706	0.733	0.850	**0.890**	**0.908**	**0.762**	**0.898**	0.789	**0.824**	**0.790**
Latin-2	**0.733**	0.763	0.792	**0.666**	0.758	0.688	**0.734**	0.854	**0.890**	0.891	0.759	0.884	0.782	0.819	0.787
Single lang.	0.725	**0.766**	**0.793**	0.611	**0.775**	0.701	0.729	0.813	0.889	0.887	0.742	0.891	0.781	0.807	0.778

3.3.2 NER through a combination of global and contextual features

We have worked on a novel hierarchical neural model for NER that uses two types of features, global and contextual.

Global features capture latent syntactic and semantic similarities. They are captured using a Graph Convolution Network (GCN) which is fed with dependencies trees.

Contextual features represent the word's semantics in a particular context. This can address aspects such as polysemy and the context-dependent nature of words. These features, obtained at sentence level, are determined through a pre-trained XLNet (Yang et al., 2019) model.

Both types of features are joined through a vector concatenation. And then, these vectors are introduced into a linear layer, which will decide the NER labels for each token in a sentence.

Although the current work has been explored only on English, with CoNLL 2003 (Tjong Kim Sang & De Meulder, 2003), the F1-score of 0.938 suggests that this method could be applied to other languages to improve their performance.

For a more detailed description of the system, see Appendix E.

4 Final Evaluation of Named Entity Linking (Task 2.1)

We present in this section our work regarding the final evaluation of the Named Entity Linking (NEL) system that we developed for WP2. Besides the evaluation, we introduce briefly this NLP tasks. As well, we show a list of annotated datatsets that are related to the NEL task.

4.1 Background

Named entity linking (NEL) is an NLP task that aims to disambiguate named entities by linking them to a knowledge base (Shen, Wang, & Han, 2014). Currently, NEL systems can be grouped into two classes: disambiguation systems, and end-to-end systems.

Disambiguation systems consist of architectures that take as input the output generated by a NER system. Then, the NEL system is charged with disambiguating each named entity and linking it to a specific knowledge-base. Some examples of disambiguation systems are (Ganea & Hofmann, 2017; Onoe & Durrett, 2020).

End-to-end systems, unlike disambiguation ones, takes as input the raw text, extract the named entities, and then disambiguate the extracted entities by linking them to a knowledge-base. Some of the end-to-end systems that can be found in the literature are (Kolitsas, Ganea, & Hofmann, 2018; Cucerzan, 2007; Broscheit, 2020; van Hooland, De Wilde, Verborgh, Steiner, & Van de Walle, 2013; Munnelly & Lawless, 2018; Ruiz & Poibeau, 2019).

Unlike NER, the number of works related to entity linking in languages different than English is reduced (Raiman & Raiman, 2018; Rijhwani, Xie, Neubig, & Carbonell, 2019; Zhou, Rijhwani, & Neubig, 2019).

4.2 Available Resources

In Table 4, we provide a list of the datasets containing annotations regarding Named Entity Linking (NEL).

Table 4: The collected corpora for the NEL task.

Name	Year	Public	Language	Location
AIDA	2003	Yes	en	link
AQUAINT	2008	Yes	en	link
ACE2004	2011	Yes	en	link
CLUEWEB	2013	Yes	en	link
MSNBC	2007	Yes	en	link
WIKIPEDIA	2011	Yes	en	link
TAC2010	2010	No	en	link
McN-dataset	2011	Yes	ar, bg, cs, da, de, el, es, fi, fr, hr, it, mk, nl, pt, ro, sq, sr, sv, tr, ur, zh	link
TAC2015	2015	No	en, es, zh	link
TH-dataset	2016	Yes	ar, de, es, fr, he, it, ta, th, tl, tr, ur, zh	link
Wikiann	2017	Yes	282 languages (et, fi, hr, lt, lv, ru, sl, sv)	link
SlavNER 2017	2017	Yes	cs, hr, pl, ru, sk, sl, uk	link
SlavNER 2019	2019	Yes	bg, cs, pl, ru	link
SlavNER 2021	2021	Yes	bg, cs, pl, ru, sl, uk	link

In the case of SlavNER, the corpora, although annotated with entity linking, the links do not point a specific knowledge-base, like Wikidata or Wikipedia. Furthermore, as indicated previously in Section 3.2, the dataset for SlavNER 2021 (Piskorski et al., 2021) (see Appendix F), was partially annotated by the Jožef Stefan Institute (JSI).

4.3 Results

In Deliverable D2.2, we proposed a cross-lingual Named Entity Linking (NEL) system based on the ideas of (Ganea & Hofmann, 2017). This work, which was published in (Linhares Pontes, Doucet, & Moreno, 2020), consists of training a NEL system using MUSE embeddings (Conneau, Lample, Ranzato, Denoyer, & Jégou, 2017) to represent words and entities in multiple languages into the same dimensional space. Therefore, the NEL system is capable of disambiguating named entities across different languages.

As the cross-lingual NEL system has multiple limitations, in Deliverable D2.5, we explored the creation of a multilingual NEL system. In this case, we used multiple probabilities tables, created specifically for each language, along with different techniques to improve the matching of candidates and entities.

In Table 5, we present the results from applying our NEL systems over the corpus WikiANN. As we can notice, the cross-lingual system has worse performance, in all languages except Slovene, than the multilingual approach.

Table 5: Micro F-score obtained by the cross-lingual NEL system (D2.2) and the multilingual one (D2.5) on the WikiANN dataset. In the case of the cross-lingual NEL system, we took the best score, either presented in D2.2 or D2.5. The improvement is calculated using Equation 1.

	Micro F1-score		Improvement
Language	**Cross-lingual**	**Multilingual**	**(%)**
et	0.584	0.769	31.520
fi	0.666	0.849	27.477
hr	0.615	0.830	34.893
lt	-	0.623	∞
lv	-	0.766	∞
ru	-	0.516	∞
sl	0.706	0.705	-0.141
sv	0.467	0.932	99.571
Average	0.607	0.748	38.664

5 Final Evaluation of Event Detection (Task 2.1)

This section presents the datasets used for the final evaluation of Event Detection (ED) as well as the results obtained from the assessment.

5.1 Background

Event extraction (EE) is an NLP task that consists of obtaining specific knowledge of certain incidents from textual documents e.g. event-related information from texts. Commonly defined in the field of IE, it consists of two main sub-tasks: event detection (ED) that deals with the extraction of critical information regarding an event, that can be represented by a keyword or a span of text, which evokes that event; and event argument extraction, concentrates on obtaining the event extents referring to more details about the events.

Over the years, several event definitions have been proposed, starting in 1991 during the Message Understanding Conferences (MUC). Due to the complexity of the initial EE task, throughout the years, it has been separated into single tasks, as NER, NEL, entity coreference, and relation extraction. Thus, the event detection task is challenging due to the ambiguous nature of the concept of event. Generally, after NER and NEL, ED takes advantage of the detected and linked named entities since they can be participants of an event.

We remind that this final deliverable does not concern the argument extraction. More explicitly, our work focused on ED. In previous deliverable, we analysed two different ways of events annotations. Moreover, we continued our work on experimenting with ED as a NER task (an event is represented as an entity, e.g., Brexit is an event) and we analysed two different annotation styles that are widely used in the research in this field.

5.2 Available Resources

We present in Table 6 the gathered datasets regarding the NLP tasks Event Dectection (ED).

Table 6: The collected datasets for the ED task.

Name	Year	Public	Language	Location
SlavNER 2019	2019	Yes	bg, cs, pl, ru	link
SlavNER 2021	2021	Yes	bg, cs, pl, ru, sl, uk	link
DanIEL	2020	No	en, fr, el, ru, zh, pl	link
ACE2005	2005	No	en	link

5.3 Results

In Deliverable D2.2, published in (Moreno et al., 2019), the same proposed NER system, based on the work of (Reimers & Gurevych, 2019), was utilised for detecting events. This BiLSTM-based architecture with BERT (Devlin et al., 2019) and FastText (Bojanowski, Grave, Joulin, & Mikolov, 2017a) embeddings was applied in the context of the SlavNER 2019 dataset that consisted of four Slavic languages: Bulgarian, Czech, Polish, and Russian. As presented in (Tsygankova, Mayhew, & Roth, 2019), there is a large imbalance in the amount of training data by language, with the largest (Polish), containing almost three times as many tokens as the smallest (Russian). The training data is in the form of newswire articles and contains document-level annotations of five different entity types: persons (PER), locations (LOC), organisations (ORG), events (EVT), and products (PRO). We, thus, compare our previously obtained results with a newly proposed model and published in (Boros, Linhares Pontes, et al., 2020) and (Boros, Hamdi, et al., 2020), with an XLM-RoBERTa encoder (Conneau et al., 2019). The XLM-RoBERTa is a Transformer-based masked cross-lingual language model trained on one hundred languages, using more than two terabytes of filtered CommonCrawl data. We chose this model due to the fact that it significantly outperformed the *bert-multilingual* on a variety of cross-lingual benchmarks.

From Table 7, the results for the event (EVT) SlavNER 2019 dataset clearly state that our proposed was more suited for this task considering that it obtained the highest F1 values for all the languages, for the EVT tag. Due to the imbalance of this tag in the documents, the BiLSTM+CNN is not able to capture enough informative features about EVT, and thus, it obtains an F1 of 0 for Czech and Russian, while all the fine-tuned pre-trained language models manage to outperform this model. The improvements, when comparing with the results in Deliverable 2.2, are of at least 39%.

Table 7: Micro F1-score regarding ED on the dataset SlavNER 2019. The BiLSTM+CNN comes from D2.2 while the XML-RoBERTa-base is described in D2.5. The improvement is calculated using Equation 1.

	Micro F1-score		Improvement
Language	**BiLSTM+CNN**	**XLM-RoBERTa-base**	**(%)**
bg	0.265	0.370	39.620
cs	0	0.680	∞
pl	0.201	0.462	129.850
ru	0	0.714	∞
Average	0.116	0.359	43.740

Moreover, in Deliverable D2.5, not only that we reported better scores for the SlavNER 2019 dataset, but we also experimented with two other datasets. ACE 2005[2] is a dataset, with various national and international events, that is widely utilised in the research community for the evaluation and comparison of IE systems and approaches, and DAnIEL dataset specialised in epidemiological events proposed by (Lejeune, Brixtel, Doucet, & Lucas, 2015). For both datasets, we proposed several models based on pre-

[2]https://catalog.ldc.upenn.edu/LDC2006T06

trained and fine-tuned language models, and we obtained state-of-the-art results that were published in (Boros et al., 2021b), (Boros et al., 2021a), (Mutuvi et al., 2020).

6 Final Evaluation of Multilingual Keyword Extraction Methods (Task T2.2)

In the scope of T2.2, we developed supervised and unsupervised methods for keyword extraction, called TNT-KID (Martinc et al., 2021) and RAKUN (Škrlj, Repar, & Pollak, 2019), respectively, that are described in Deliverables D2.2 and D2.6. In D2.6 we also presented initial experiments for improving the recall, for which the final evaluation is provided in this deliverable.

The contribution in terms of datasets consists of media partners datasets with train and test splits used for keyword extraction. The datasets have been briefly introduced in D2.6, but since then, we released them publicly on CLARIN.

6.1 Background

Many different approaches have been developed to tackle the problem of extracting keywords. The early approaches, such as KP-MINER (El-Beltagy & Rafea, 2009) and RAKE (Rose, Engel, Cramer, & Cowley, 2010) rely on unsupervised techniques which employ frequency-based metrics for extraction of keywords from text. Most recent state-of-the-art statistical approaches, such as YAKE (Campos et al., 2018), also employ frequency-based features, but combine them with other features such as casing, position, relatedness to context, and dispersion of a specific term in order to derive a final score for each keyword candidate.

Another line of research models this problem by exploiting concepts from graph theory. Approaches, such as TextRank (Mihalcea & Tarau, 2004), Single Rank (Wan & Xiao, 2008), TopicRank (Bougouin, Boudin, & Daille, 2013) and Topical PageRank (Sterckx, Demeester, Deleu, & Develder, 2015) build a graph G, i.e. a mathematical construct described by a set of vertexes V and a set of edges E connecting two vertices. In one of the most recent approaches developed during the project, RaKUn (Škrlj et al., 2019), a directed graph is constructed from text, and keywords are ranked by a shortest path-based metric from graph theory - the load centrality.

The task of keyword extraction can also be tackled in a supervised way. One of the first supervised approaches was an algorithm named KEA (Witten, Paynter, Frank, Gutwin, & Nevill-Manning, 2005), which uses only TF-IDF and the term's position in the text as features for term identification. More recent neural approaches to keyword detection consider the problem as a sequence-to-sequence generation task (Meng et al., 2017).

Finally, the newest branch of models considers keyword extraction as a sequence labelling task and tackles keyword detection with Transformers. (Sahrawat et al., 2020) fed contextual embeddings generated by several Transformer models including BERT (Devlin, Chang, Lee, & Toutanova, 2018), RoBERTa (Liu et al., 2019), GPT-2 (Radford et al., 2019), etc.) into two types of neural architectures, a bidirectional Long short-term memory network (BiLSTM) and a BiLSTM network with an additional Conditional random fields layer (BiLSTM-CRF). (Sun, Xiong, Liu, Liu, & Bao, 2020) on the other hand proposed BERT-JointKPE that employs a chunking network to identify phrases and a ranking network to learn their salience in the document. By training BERT jointly on the chunking and ranking tasks the model manages to establish a balance between the estimation of keyphrase quality and salience.

We contributed a state-of-the-art Transformer-based approach TNT-KID (Transformer-based Neural Tagger for Keyword Identification) (Martinc et al., 2021), which does not rely on pre-trained language models such as BERT, but rather allows the user to train their own language model on the appropriate domain. The study shows that smaller unlabelled domain-specific corpora can be successfully used

for unsupervised pretraining, which makes the proposed approach easily transferable to low-resource languages. It also proposes several modifications to the Transformer architecture in order to adapt it for a keyword extraction task and improve the performance of the model.

While in D2.3 and D2.6 we presented novel methods developed, the unsupervised RaKUN (Škrlj et al., 2019) and the supervised TNT-KID (Martinc et al., 2021) and evaluated them on public datasets, this deliverable focuses primarily on evaluation on media partners' datasets, where only initial monolingual results were provided in D2.6. We present the final experiments on these tasks and evaluate the methods also in multilingual and cross-lingual settings.

6.2 Methods

The TNT-KID model developed during the project (T2.2), as well as the BERT-based approach by (Sahrawat et al., 2020) that we use in our cross-lingual and multilingual experiments are briefly described below.

6.2.1 TNT-KID

TNT-KID (Martinc et al., 2021), described in detail in D2.6, is a supervised model leveraging the Transformer architecture (Vaswani et al., 2017), which was somewhat adapted for the specific task at hand. It does not rely on pre-trained language models such as BERT, but rather allows the user to train their own language model on the appropriate domain. For each dataset, we first pre-train the model with an auto-regressive language model objective. After that, the model is fine-tuned on the train set for the keyword extraction task.

6.2.2 BERT+ BiLSTM-CRF

Another keyword extraction approach we employ in our work is the method proposed by (Sahrawat et al., 2020), where they fed contextual embeddings generated by BERT into a BiLSTM network with an additional Conditional random fields layer (BiLSTM-CRF). This state-of-the-art supervised approach is used in our cross-lingual experiments and is also used as a baseline in the experiments presented in Section 6.2.3

6.2.3 TF-IDF(tm): Extending supervised keyword extraction by TF-IDF tagset matching

The recent supervised neural methods are very precise, but in some cases, they do not return a sufficient number of keywords. This is due to the fact that the methods are trained on the training data with a low number of gold standard keywords (as it can be seen from Table 9). To meet the media partners' needs, we designed a method that complements state-of-the-art neural methods (the TNT-KID method (Martinc et al., 2021) and the Transformer-based method proposed by (Sahrawat et al., 2020), which are both described above) by a tagset matching approach, returning a constant number of keywords (k=10). More specifically, we propose a TF-IDF tagset matching technique, which finds additional keyword candidates by ranking the words in the news article that have appeared in the predefined keyword set containing words from the gold standard train set. The new hybrid system first checks how many keywords were returned by the supervised approach and if the number is smaller than needed, the list is expanded by the best-ranked keywords returned by the TF-IDF based tagset matching extraction system (the method was already presented in D2.6, but this deliverable covers final evaluation, which also resulted in the EACL workshop publication (Koloski, Pollak, Škrlj, & Martinc, 2021), presented in Appendix G.

6.2.4 Supervised keyword extraction in a multilingual and cross-lingual settings

In this deliverable, we also present a novel cross-lingual keyword extraction evaluation setting. The idea is to pre-train the model on a multilingual corpus, fine-tune it on one or more languages and then conduct cross-lingual testing of the model on languages appearing in the train set and also on languages not appearing in the train set (zero-shot transfer). The main aims of these experiments are the following:

- We want to test whether adding additional training data from languages not included in the dataset on which the model is employed, can improve the inference of the model.
- We want to test how well does the model perform in the zero-shot setting, where the model is trained only on the languages not included in the test set. Achieving a satisfactory performance in this setting would make the model transferable even to languages with no manually labeled resources.

We employ the BERT+BiLSTM-CRF method in these multilingual and cross-lingual settings with different combinations of languages in training and test sets. The method itself is the same as the one presented in (Sahrawat et al., 2020), with the exception that we used a multilingual BERT model pre-trained on about 100 Wikipedia languages (Devlin et al., 2019) to allow for cross-lingual transfer.

6.3 Resources

While RaKUN and TNT-KID have been evaluated on standard keyword extraction datasets in English, in the scope of this task, we gathered new benchmark datasets for keyword extraction that we also published on CLARIN.

The description of public datasets for keyword extraction in English has already been presented in previous deliverables of tasks T2.2 and T2.4. and are also described in our journal paper (Martinc et al., 2021). Here we only briefly summarise the datasets from the computer science domain (see Table 8), which we use for average comparison of the state-of-the-art TNT-KID method developed during the EMBEDDIA project with the pre-EMBEDDIA state-of-the-art (Section 6.4.1).

We nevertheless describe in more detail three publicly available English datasets from the news domain that we use in our experiments:

- **KPTimes (Gallina, Boudin, & Daille, 2019)**: The corpus contains 279,923 news articles containing editor assigned keywords that were collected by crawling New York Times news website[3]. After that, the dataset was randomly divided into training (92.8%), development (3.6%), and test (3.6%) sets.
- **JPTimes (Gallina et al., 2019)**: Similar as **KPTimes**, the corpus was collected by crawling Japan Times online news portal[4]. The corpus only contains 10,000 English news articles and is used in our experiments as a test set for the classifiers trained on the **KPTimes** dataset.
- **DUC (Wan & Xiao, 2008)**: The dataset consists of 308 English news articles and contains 2,488 hand labeled keyphrases.

The statistics about the datasets that are used for training and testing of our models are presented in Table 8. Note that there is a big variation in dataset sizes in terms of number of documents (column *No. docs*), and in an average number of keywords (column *Avg. kw.*) and present keywords per document (columns *Avg. present kw.*), ranging from 2.35 present keywords per document in *KPTimes-valid* to 7.79 in *DUC-test*.

[3] https://www.nytimes.com
[4] https://www.japantimes.co.jp

Table 8: Datasets used for empirical evaluation of keyword extraction algorithms. *No.docs* stands for number of documents, *Avg. doc. length* stands for average document length in the corpus (in terms of the number of words, i.e., we split the text by white-space), *Avg. kw.* stands for the average number of keywords per document in the corpus, *% present kw.* stands for the percentage of keywords that appear in the corpus (i.e., percentage of document's keywords that appear in the text of the document) and *Avg. present kw.* stands for the average number of keywords per document that actually appear in the text of the specific document.

Dataset	No. docs	Avg. doc. length	Avg. kw.	% present kw.	Avg. present kw.
Computer science papers					
KP20k-train	530,000	156.34	5.27	62.43	3.29
KP20k-valid	20,000	156.55	5.26	62.30	3.28
KP20k-test	20,000	156.52	5.26	62.55	3.29
Inspec-valid	1,500	125.21	9.57	76.92	7.36
Inspec-test	500	121.82	9.83	78.14	7.68
Krapivin-valid	1,844	156.65	5.24	54.34	2.85
Krapivin-test	460	157.76	5.74	55.66	3.20
NUS-test	211	164.80	11.66	50.47	5.89
SemEval-valid	144	166.86	15.67	45.43	7.12
SemEval-test	100	183.71	15.07	44.53	6.71
News articles					
KPTimes-train	259,923	783.32	5.03	47.30	2.38
KPTimes-valid	10,000	784.65	5.02	46.78	2.35
KPTimes-test	10,000	783.47	5.04	47.59	2.40
JPTimes-test	10,000	503.00	5.03	76.73	3.86
DUC-test	308	683.14	8.06	96.62	7.79

EMBEDDIA project released also **novel keyword extraction datasets**, consisting of news articles and corresponding keywords that were assigned by the journalists. These datasets were released as part of the EMBEDDIA resources (Pollak et al., 2021) proposed in the scope of the EACL Hackashop on News Media Content Analysis and Automated Report Generation (Toivonen & Boggia, 2021). The datasets cover news in four languages; Latvian, Estonian, Russian, and Croatian. Latvian, Estonian, and Russian datasets contain news from the Ekspress Group, specifically from Estonian Ekspress Meedia (news in Estonian and Russian) and from Latvian Delfi (news in Latvian and Russian). The Croatian dataset was acquired from 24sata news portal belonging to Styria Media Group. The dataset statistics and their train/test splits are presented in Table 9 and released on CLARIN[5] and described in detail in (Koloski, Pollak, Škrlj, & Martinc, 2021)). From the news articles made available by media houses (Pollak et al., 2021), for Latvian, Estonian, and Russian, we selected the articles from 2018 for the training set, while for the test set the articles from 2019 were used. For Croatian, the articles from 2019 are arranged by date and split into training and test (i.e., about 10% of the 2019 articles with the most recent date) set.

In our study (Section 6.2.3), we also use tagsets of keywords. Tagset corresponds either to a collection of keywords maintained by editors of a media house (see e.g., Estonian tagset) or to a tagset constructed from assigned keywords from articles available in the training set. The type of tagset and the number of unique tags for each language are listed in Table 10.

6.4 Experiments and results

Next, we describe the experiments and results obtained.

[5] https://www.clarin.si/repository/xmlui/handle/11356/1403

Table 9: Media partners' datasets used for empirical evaluation of keyword extraction algorithms.

			Avg. Train					Avg. Test				
Dataset	**Total docs**	**Total kw.**	**Total docs**	**Doc len**	**Kw.**	**% present kw.**	**present kw.**	**Total docs**	**Doc len**	**Kw.**	**% present kw.**	**Present kw.**
Croatian	35,805	126,684	32,223	438.50	3.54	0.32	1.19	3582	464.39	3.53	0.34	1.26
Estonian	18,497	59,242	10,750	395.24	3.81	0.65	2.77	7,747	411.59	4.09	0.69	3.12
Russian	25,306	5,953	13,831	392.82	5.66	0.76	4.44	11,475	335.93	5.43	0.79	4.33
Latvian	24,774	4,036	13,133	378.03	3.23	0.53	1.69	11,641	460.15	3.19	0.55	1.71

Table 10: Distribution of tags provided per language. The media houses provided tagsets for Estonian and Russian, while the tags for Latvian and Croatian were extracted from the train set.

Dataset	Unique tags	Type of tags
Croatian	21,165	Constructed
Estonian	52,068	Provided
Russian	5,899	Provided
Latvian	4,015	Constructed

6.4.1 Comparing supervised EMBEDDIA methods with pre-EMBEDDIA state-of-the art methods on public datasets

In D2.6 we presented a novel supervised approach for keyword extraction, TNT-KID, published the journal paper (Martinc et al., 2021) containing also a detailed comparison with other state-of-the-art approaches.

To better evaluate the contribution of the EMBEDDIA project to the field of keyword extraction in general and to assess the improvement over previous pre-EMBEDDIA state-of-the-art, this section we offer a comparison between TNT-KID and a pre-EMBEDDIA state-of-the-art keyword extraction method CopyRNN (Meng et al., 2017), which employs a generative model for keyword prediction with a recurrent encoder-decoder framework with an attention mechanism capable of detecting keywords in the input text sequence and also potentially finding keywords that do not appear in the text. In Table 11, we compare the methods on three publicly available news datasets described in Section 6.3 and in terms of average performance across nine publicly available datasets used for evaluation of TNT-KID in the original study (Martinc et al., 2021).

Table 11: Comparison between the previous pre-EMBEDDIA state-of-art method for supervised keyword extraction CopyRNN and the proposed TNT-KID approach on three English news datasets and on average across nine datasets (six computer science paper datasets and three news datasets) on which the TNT-KID approach was tested in terms of F1@5 and F1@10.

Dataset	CopyRNN	TNT-KID	Improvement (%)
KPTimes F1@5	0.406	0.485	19.46
KPTimes F1@10	0.393	0.485	23.41
JPTimes F1@5	0.246	0.359	45.93
JPTimes F1@10	0.256	0.361	41.02
DUC F1@5	0.083	0.318	283.13
DUC F1@10	0.105	0.373	255.24
Average F1@5	0.288	0.363	26.04
Average F1@10	0.280	0.389	38.93

The improvement is the biggest of on the DUC dataset (283.13% and 255% in terms of F1@5 and F1@10, respectively), which is much smaller than the other two news datasets. This indicates that

the proposed TNT-KID algorithm is especially useful for extracting keywords on datasets too small for the training of the CopyRNN model. The improvement is also substantial on the much larger JPTimes dataset (Gallina et al., 2019) containing 10,000 English news articles from Japan Times, where we manage to improve on the pre-EMBEDDIA state-of-the-art by 45.93% in terms of F1@5 and 41.02% in terms of F1@10. Improvements on the KPTimes datasets (Gallina et al., 2019) are roughly two times smaller. Across all nine English datasets, on which TNT-KID was tested (six from the computer science domain and three from the news domain), we achieve an average improvement of 26.04% in terms of F1@5 and 38.93% in terms of F1@10.

6.4.2 Evaluation of keyword extractors on EMBEDDIA datasets in a monolingual setting

In this section, we present the final evaluation of keyword extraction methods on the four novel EMBEDDIA media partners' datasets, described in Section 6.3. While methods were implemented in the scope of T2.3 and similar results presented in D2.6, this deliverable presents the final evaluation on the train-test splits released on CLARIN (Koloski, Pollak, Škrlj, & Martinc, 2021)[6]. In addition, in the previous deliverable, some of the methods were not applied to Russian and Latvian, and here we provide complete results. The work presented in this section is also published in (Koloski, Pollak, Škrlj, & Martinc, 2021), provided as Appendix G of this deliverable.

The experiments performed in this section have two main contributions. First, we compare the results of two supervised neural methods as well as their combination, and second, we address the need identified by our media partners, to improve the recall and return a constant number of keywords by our method for tagset matching.

We evaluate the following methods and combinations of methods, which are described in Section 6.2 and applied in the following way:

- **TF-IDF(tm):** TF-IDF-based weighting of keywords from the tagset is used, and the top-ranked keywords that are present in the tagset are selected. For details see (Koloski, Pollak, Škrlj, & Martinc, 2021), and the summary provided in in Section 6.2.3.
- **TNT-KID** (Martinc et al., 2021): For each dataset, we first pre-train the model with an autoregressive language model objective. After that, the model is fine-tuned on the same train set for the keyword extraction task. Sequence length was set to 256, embedding size to 512, and batch size to 8, and we employ the same preprocessing as in the original study (Martinc et al., 2021).
- **BERT + BiLSTM-CRF** (Sahrawat et al., 2020): We employ an uncased multilingual BERT[7] model with an embedding size of 768 and 12 attention heads, with an additional BiLSTM-CRF token classification head, same as in (Sahrawat et al., 2020).
- **TNT-KID & BERT + BiLSTM-CRF**: We extracted keywords with both of the methods and complemented the TNT-KID extracted keywords with the BERT + BiLSTM-CRF extracted keywords in order to retrieve more keywords. Duplicates (i.e., keywords extracted by both methods) are removed.
- **TNT-KID & TF-IDF(tm)**: If the keyword set extracted by TNT-KID contains less than 10 keywords, it is expanded with keywords retrieved with the proposed TF-IDF(tm) approach explained above, which do not appear in the keyword set extracted by TNT-KID.
- **BERT + BiLSTM-CRF & TF-IDF(tm)**: If the keyword set extracted by BERT + BiLSTM-CRF contains less than 10 keywords, it is expanded with keywords retrieved with the proposed TF-IDF(tm) approach, i.e., best-ranked keywords according to TF-IDF, which do not appear in the keyword set extracted by BERT + BiLSTM-CRF.

[6] http://hdl.handle.net/11356/1403

[7] More specifically, we use the 'bert-base-multilingual-uncased' implementation of BERT from the Transformers library (https://github.com/huggingface/transformers).

Table 12: Results on the EMBEDDIA media partner datasets.

Model	P@5	R@5	F1@5	P@10	R@10	F1@10
Croatian						
TF-IDF	0.2226	0.4543	0.2988	0.1466	0.5888	0.2347
TNT-KID	0.3296	0.5135	0.4015	0.3167	0.5359	0.3981
BERT + BiLSTM-CRF	**0.4607**	0.4672	**0.4640**	**0.4599**	0.4708	**0.4654**
TNT-KID & TF-IDF(tm)	0.2659	0.5670	0.3621	0.1688	0.6944	0.2716
BERT + BiLSTM-CRF & TF-IDF(tm)	0.2644	0.5656	0.3604	0.1549	0.6410	0.2495
TNT-KID & BERT + BiLSTM-CRF	0.2940	0.5447	0.3820	0.2659	0.5968	0.3679
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.2648	**0.5681**	0.3612	0.1699	**0.7040**	0.2738
Estonian						
TF-IDF	0.0716	0.1488	0.0966	0.0496	0.1950	0.0790
TNT-KID	**0.5194**	0.5676	**0.5424**	**0.5098**	0.5942	**0.5942**
BERT + BiLSTM-CRF	0.5118	0.4617	0.4855	0.5078	0.4775	0.4922
TNT-KID & TF-IDF(tm)	0.3463	0.5997	0.4391	0.1978	0.6541	0.3037
BERT + BiLSTM-CRF & TF-IDF(tm)	0.3175	0.4978	0.3877	0.1789	0.5381	0.2686
TNT-KID & BERT + BiLSTM-CRF	0.4421	0.6014	0.5096	0.4028	0.6438	0.4956
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.3588	**0.6206**	0.4547	0.2107	**0.6912**	0.3230
Russian						
TF-IDF	0.1764	0.2314	0.2002	0.1663	0.3350	0.2223
TNT-KID	**0.7108**	0.6007	**0.6512**	**0.7038**	0.6250	**0.6621**
BERT + BiLSTM-CRF	0.6901	0.5467	0.5467	0.6849	0.5643	0.6187
TNT-KID & TF-IDF(tm)	0.4519	0.6293	0.5261	0.2981	0.6946	0.4172
BERT + BiLSTM-CRF & TF-IDF(tm)	0.4157	0.5728	0.4818	0.2753	0.6378	0.3846
TNT-KID & BERT + BiLSTM-CRF	0.6226	0.6375	0.6300	0.5877	0.6707	0.6265
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.4622	**0.6527**	0.5412	0.2965	**0.7213**	0.4203
Latvian						
TF-IDF	0.2258	0.5035	0.3118	0.1708	0.5965	0.2655
TNT-KID	0.6089	0.6887	**0.6464**	0.6054	0.6960	**0.6476**
BERT + BiLSTM-CRF	**0.6215**	0.6214	0.6214	**0.6204**	0.6243	0.6223
TNT-KID & TF-IDF(tm)	0.3402	**0.7934**	0.4762	0.2253	0.8653	0.3575
BERT + BiLSTM-CRF & TF-IDF(tm)	0.2985	0.6957	0.4178	0.1889	0.7427	0.3012
TNT-KID & BERT + BiLSTM-CRF	0.4545	0.7189	0.5569	0.4341	0.7297	0.5443
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.3318	0.7852	0.4666	0.2124	**0.8672**	0.3414

- **TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm):** the keyword set extracted with the TNT-KID is complemented by keywords extracted with BERT + BiLSTM-CRF (duplicates are removed). If after the expansion the keyword set still contains less than 10 keywords, it is expanded again, this time with keywords retrieved by the TF-IDF(tm) approach.

For TNT-KID, which is the only model that requires language model pretraining, language models were trained on train sets in Table 9 for up to ten epochs. Next, TNT-KID and BERT + BiLSTM-CRF were fine-tuned on the training datasets, which were randomly split into 80 percent of documents used for training and 20 percent of documents used for validation. The documents containing more than 256 tokens are truncated, while the documents containing less than 256 tokens are padded with a special $<$ pad $>$ token at the end. We fine-tuned each model for a maximum of 10 epochs and after each epoch, the trained model was tested on the documents chosen for validation. The model that showed the best performance on this set of validation documents (in terms of F1@10 score) was used for keyword detection on the test set.

For evaluation, we employ precision, recall, and F1-score. While F1@10 and recall@10 are the most rel-

evant metrics for the media partners, we also report precision@10, precision@5, recall@5, and F1@5. Only keywords that appear in a text (present keywords) were used as a gold standard since we only evaluate approaches for keyword tagging that are not capable of finding keywords that do not appear in the text. Lowercasing and lemmatization (stemming in the case of Latvian) are performed on both the gold standard and the extracted keywords (keyphrases) during the evaluation. The results of the evaluation on all four languages are listed in Table 12.

The results suggest that neural approaches, TNT-KID and BERT+BiLSTM-CRF, offer comparable performance on all datasets but nevertheless achieve different results for different languages. TNT-KID outperforms BERT-BiLSTM-CRF model according to all the evaluation metrics on the Estonian and Russian news datasets. It also outperforms all other methods in terms of precision and F1-score. On the other hand, BERT+BiLSTM-CRF performs better on the Croatian dataset in terms of precision and F1-score. On Latvian, TNT-KID achieves top results in terms of F1, while BERT+BiLSTM-CRF offers better precision.

Even though the TF-IDF tagset matching method performs poorly on its own, we can nevertheless observe that we can drastically improve the recall@5 and the recall@10 of both neural systems, if we expand the keyword tag sets returned by the neural methods with the TF-IDF ranked keywords from the tagset. The improvement is substantial and consistent for all datasets, but it nevertheless comes at the expanse of the lower precision and F1-score. This is not surprising, since the final expanded keyword set always returns 10 keywords, i.e., much more than the average number of present gold standard keywords in the media partner datasets (see Table 9), which badly affects the precision of the approach. Nevertheless, since for a journalist a manual inspection of 10 keyword candidates per article and manual selection of good candidates (e.g., by clicking on them) still requires less time than the manual selection of keywords from an article, we argue that the improvement of recall at the expanse of the precision is a good trade-off if the system is intended to be used as a recommendation system in the media house environment.

Combining keywords returned by TNT-KID and BERT + BiLSTM-CRF also consistently improves recall, but again at the expanse of lower precision and F1-score. Overall, for all four languages, the best performing method in terms of recall is the TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm).

Finally, we can also compare the TNT-KID approach with the current state-of-the-art method for keyword extraction BERT+BiLSTM-CRF (Sahrawat et al., 2020) in terms of F1@10. We achieve performance improvements of 20.7% for Estonian, 7% for Russian, and 4.1% for Latvian. Interestingly, we observe a decrease in performance of 16.9% on the Croatian dataset. This is most likely due to the different keyword labelling regime employed by the Styria group media partner (i.e. fewer present keywords per article). Overall, the average improvement across all four media partner datasets is 3.7%.

6.4.3 Comparing supervised keyword extractors on EMBEDDIA datasets in multilingual and cross-lingual settings

In this setting, we only use a multilingual BERT + BiLSTM-CRF model (Sahrawat et al., 2020) due to its multilingual language model pre-training, which allows the zero-shot transfer of model's knowledge across languages[8]. The model has an embedding size of 768 and 12 attention heads. We add an additional BiLSTM-CRF token classification layer on top of the model, same as in (Sahrawat et al., 2020).

In order to explore the applicability of the proposed approach in multilingual and cross-lingual settings, we utilise the following approach: For each given test language l we train two different classification models, a cross-lingual model trained on three languages not appearing in the test set, and a multilingual model trained on the dataset consisting of all four media partner train sets. More formally, in a cross-lingual setting, we fine-tune the previously described model on all the languages from the set $L-l$, where

[8]More specifically, we use the 'bert-base-multilingual-uncased' implementation of BERT from the Transformers library (`https://github.com/huggingface/transformers`).

l is a given evaluation language, and L is a set of four languages $L = \{Croatian, Latvian, Russian, Estonian\}$. In the multilingual setting, we train a model on a multilingual space consisting of all of the training sets of languages in L and then employ this model on each of the four languages' test sets.

The results are presented in Table 13, where the column *type* denotes in what setting was the model evaluated, **cross** for cross-lingual or **multi** for multilingual. A monolingual setting (type **mono**) is used as a baseline, in which the model was trained on the train set with a language corresponding to the language of the test set (see (Koloski, Pollak, Škrlj, & Martinc, 2021) for details). The cross-lingual results allow for estimation of the performance of the model on a new language, while the multilingual setting explores if training data in other languages can lead to improvement of results.

Table 13: Evaluation of the cross-lingual and multilingual approach to keyword extraction. The column *type* denotes in what setting was the model evaluated, **mono** for monolingual, **cross** for cross-lingual or **multi** for multilingual setting.

Model	Type	P@5	R@5	F1@5	P@10	R@10	F1@10
Evaluation on Croatian test set							
Croatian	mono	0.4607	0.4672	0.464	0.464	0.4708	0.4654
Estonian-Latvian-Russian	cross	0.1245	0.0877	0.103	0.1244	0.0877	0.103
Croatian-Estonian-Latvian-Russian	multi	0.3334	0.3179	0.3255	0.3329	0.3186	0.3256
Evaluation on Estonian test set							
Estonian	mono	0.5118	0.4617	0.4855	0.5078	0.4775	0.4922
Croatian-Latvian-Russian	cross	0.3291	0.2405	0.278	0.3291	0.2413	0.2785
Croatian-Estonian-Latvian-Russian	multi	0.4561	0.4068	0.4301	0.4548	0.4166	0.4349
Evaluation on Latvian test set							
Latvian	mono	0.6215	0.6214	0.6124	0.6204	0.6243	0.6223
Croatian-Estonian-Russian	cross	0.2227	0.2337	0.2281	0.2220	0.2346	0.2282
Croatian-Estonian-Latvian-Russian	multi	0.5102	0.4779	0.4936	0.5098	0.4797	0.4943
Evaluation on Russian test set							
Russian	mono	0.6901	0.5467	0.5467	0.6849	0.5643	0.6187
Croatian-Estonian-Latvian	cross	0.235	0.1753	0.2008	0.2348	0.1781	0.2026
Croatian-Estonian-Latvian-Russian	multi	0.6821	0.4881	0.569	0.6804	0.4991	0.5759

Interestingly, in no language, the proposed multilingual models outperform the models built on a single language. This indicates that the additional foreign language information in the multilingual dataset mainly introduces noise in the model and does not offer a lot of useful information, that the model would not obtain during monolingual training. On the other hand, the cross-lingual zero-shot keyword extraction shows rather promising results for some languages. For example, on the Estonian test set, the cross-lingual model obtains an F1@10 of 27.85%. While this is a much lower score than the score achieved by a monolingual model (F1@10 of 49.22%), it still indicates that the zero-shot keyword extraction is nevertheless possible and should be explored more thoroughly in future work.

7 Final evaluation of Term Extraction (Task T2.2)

In this section, we focus on the term extraction, where we present the evaluation of neural models on the ACTER dataset (Rigouts Terryn et al., 2019), which is the current benchmark dataset for the task. In D2.6, we proposed an initial approach, where results from a statistical method were re-ranked using a score from the differences of term contextual embeddings in a domain and reference corpus (using ELMo embeddings). However, in this final evaluation, we opted for a different method, i.e. modelling

the term extraction as a sequence modelling task and comparing different transformer models, which is much faster, can be easily tested on different languages, and improved the score over the method described in D2.6. In relation to keywords, terms are considered on the collection level, not on a single document level. In terms of news-related analysis, the terms would be interesting for comparing core vocabulary for specific genres.

7.1 Background

Terminology extraction is an NLP task that eases the effort of manually identifying terms from domain-specific corpora by providing a list of candidate terms. While keywords are meaningful on a document level, the terms are important words or word phrases on a document collection level.

Traditionally, there were two different approaches to monolingual term extraction: linguistic and statistical. The linguistic approach utilizes the distinctive linguistic aspects of terms—most often their syntactic patterns, while the statistical approach takes advantage of term frequencies in the corpus. However, most systems are hybrid, using a combination of the two approaches; e.g., (Justeson & Katz, 1995) first define part-of-speech patterns of terms and then use simple frequencies to filter the term candidates.

Many terminology extraction algorithms are based on the concepts of termhood and unithood defined by (Kageura & Umino, 1996). Termhood is “the degree to which a stable lexical unit is related to some domain-specific concepts” and unithood is “the degree of strength or stability of syntagmatic combinations and collocations”. Termhood-based statistical measures (Vintar, 2010) function on a presumption that a term’s relative frequency will be higher in domain-specific corpora than in the general language, while common statistical measures, such as mutual information (Daille, Gaussier, & Langé, 1994), are used to measure unithood. These two approaches have been used as a basis of several hybrid systems, such as TermEnsembler (Repar, Podpečan, Vavpetič, Lavrač, & Pollak, 2019) and Termolator (Meyers et al., 2018).

Most recently, the advances in embeddings and deep neural networks development have also influenced the terminology extraction field, which also represents the winning approached in the TermEval2020 competition (Rigouts Terryn, Hoste, Drouin, & Lefever, 2020). For English and French, the winning approach is by TALN-LS2N (Hazem, Bouhandi, Boudin, & Daille, 2020) who use a BERT model in a binary classification setting, where a combination of n-grams and a sentence are used as an instance. The winning approach for Dutch described in (Rigouts Terryn et al., 2020) on the other hand use pre-trained GloVe word embeddings that are fed into a bidirectional LSTM based neural architecture.

7.2 Available resources

For multilingual term extraction, the ACTER corpus represents the best resource for evaluating and comparing methods to the state of the art, and we use it also in our evaluation.

The ACTER corpus (Rigouts Terryn et al., 2019) contains manually annotated term candidates, and was also used as a gold standard in the TermEval2020 competition organised in the scope of .

ACTER is a manually annotated dataset for term extraction, covering trilingual languages (English, French, and Dutch), and 4 domains (corruption, dressage or equitation, heart failure, and wind energy). The size of the corpus ranges from approximately 45 thousand tokens for the *heart failure* domain to around 315 thousand tokens for the *wind energy* domain.

Four labels were used for term annotation: Specific Terms, Common Terms, Out-of-Domain Terms, and Named Entities. No preprocessing (i.e. lemmatisation) of the texts was undertaken, there were no restrictions in terms of morphosyntactic patterns or length, and all individual occurrences of terms were annotated. The statistics for each language (where all terms regardless of the categories are considered) are presented in Table 14.

Table 14: Number of tokens and unique annotated terms in each domain per language in the ACTER corpus.

Dataset	Lang	Tokens	Terms
corruption	en	176,314	1174
	fr	196,327	1217
	nl	184,541	1295
dressage	en	102,654	1575
	fr	109,572	1183
	nl	103,851	1546
heart failure	en	45,788	2585
	fr	46,751	2423
	nl	47,888	2257
wind energy	en	314,618	1534
	fr	314,681	968
	nl	308,742	1245

7.3 Method

We consider the problem of terminology extraction as a sequence labeling task, which means the model returns a label for each token. To do that, we map the terms from the gold-standard list to the tokens inside raw text (see example in Table 15) and annotate each word inside the text sequence with one of the following three labels:

- B: the word is the beginning word in the term,
- I: the word is inside the term,
- O: the word is not inside the term.

Table 15: An example of our target labels for terminology extraction.

Sent_ids	Words	Labels
...	...	. ...
3	greco	O
3	is	O
3	the	O
3	most	O
3	inclusive	O
3	existing	O
3	anti	B
3	-	I
3	corruption	I
3	monitoring	B
3	mechanism	I
...	...	. ...

We test several different Transformer-based pre-trained language models (Vaswani et al., 2017) for the task at hand, namely XLNet (Yang et al., 2019), BERT (Devlin et al., 2019), DistilBERT (Sanh, Debut, Chaumond, & Wolf, 2019) and RoBERTa (Liu et al., 2019). Each model is fine-tuned on the training set to predict the probability for each word in a word sequence that a word is a part of a term (B,I) or not

(O). To do that, an additional token classification head containing a feed-forward layer with a Softmax activation is added on top of each model.

We also propose several additional techniques to improve the labelling:

- **Class weighting**: As the distribution of labels is significantly imbalanced (i.e., most words in the text are not terms), we customize the class weights based on the train set label distribution[9] to reduce the effect of imbalance and avoid overfitting on the majority label O (not a term).
- **Rule based term expansion and correction**: The initial experiments suggested that the model's inference still faces some issues:
  - In many cases, the model only predicts only a part of the multi-word term (e.g., the model predicts "acute ischemic" and "heart failure", when the full gold standard term is "acute ischemic right heart failure")
  - The model cannot predict the terms longer than 5 words whereas in the ground truth list there are terms of length up to 10 and named entities of length up to 25.
  - The model mistakenly takes the punctuation such as hyphens as a single term.

  Besides removing all terms consisting of only punctuation, we also alleviate the above-described issue of predicting only part of the multi-word term. We try to increase the average length of a predicted term with an additional post-processing step, which takes advantage of the term POS patterns defined in the terminological database of the EU institutions – IATE[10]. Table 16 presents the list of most common IATE patterns with the information about the pattern ratio in the IATE termbase and the headword position inside a pattern (0 means the first word in the pattern is the headword). For example, "NOUN NOUN 0.07 0" means that 7% of all terms in the IATE term base has this pattern and that the first noun is the headword.

Table 16: Most common POS patterns of the multi-word terms in the IATE term base, their frequency of appearance and the position of the headword.

Patterns	IATE(%)	Headword
ADJ NOUN	0.14	1.
NOUN NOUN	0.07	0
ADJ ADJ NOUN	0.04	2
NOUN ADJ NOUN	0.03	2
NOUN ADP NOUN	0.03	0
NOUN ADP ADJ NOUN	0.02	0
ADJ NOUN NOUN	0.02	1
ADJ NOUN ADP NOUN	0.01	1
NOUN ADP NOUN NOUN	0.01	0
ADJ NOUN ADP ADJ NOUN	0.01	1
NOUN PROPN	0.01	0
NOUN NOUN NOUN	0.01	0
ADV ADJ NOUN	0.01	2
ADJ NOUN ADJ NOUN	0.01	1

We analyse how expansion by different IATE patterns influences the performance of the models, where term expansion is performed in the following way: if the model predicted that a single word in the text sequence is a term, the part-of-speech (POS) tag for the predicted term was first determined. After that, the neighborhood of the word is expected, in order to determine if the neighboring words fit a specific

[9] We employ the Sklearn class estimation utility: `https://scikit-learn.org/stable/modules/generated/sklearn.utils.class_weight.compute_class_weight.html`

[10] `https://iate.europa.eu/home`

IATE pattern. If they do, the initial term is expanded. For example, if the model predicts a NOUN word as a term, we check whether the POS tag of the preceding word is an adjective (ADJ). If it is, the initial NOUN term is expanded and becomes a multi-word “ADJ NOUN” term. The initial term (NOUN) and expended terms are returned as a result.

7.4 Experiments and Results

The experiments were conducted on the ACTER datasets described in Section 7.2. As ACTER dataset contains 4 domains with 3 different languages, for each language we use 3 domains (corruption, equitation, wind) as training and validation data and the last domain of heart failure as testing data with 2 different gold standard lists: term list (ANN), and term and named entity list (NES). We randomly split the text from 3 domains (corruption, equitation, and wind) into 85% of paragraphs for training and the rest for validation. We experimented with different input sequence lengths ranging from 64 to 512 tokens and finally chose the configuration with a max length of 512 tokens, which performed the best on the validation set, for employment on the test set. The paragraphs containing more than 512 tokens are truncated, while the ones containing less than 512 tokens are padded with a special <PAD> token at the end. We fine-tuned each model for a maximum of 4 epochs and after each epoch, the trained model was tested on the documents chosen for validation.

For evaluation, we measure micro-averaged Precision, Recall, and F1-score to compare our predicted candidate term lists with the gold standard. Lowercasing and punctuation are conducted on both the gold standard and the extracted terms during the evaluation, same as in (Rigouts Terryn et al., 2020).

7.4.1 Evaluation of class weighting and term expansion techiques

For evaluating the effect of class weighting and rule-based term expansion, we used the English dataset only. We experiment with uncased and cased versions of English and multilingual BERT, RoBERTa, cased and uncased version of DistilBERT, and XLNet with and without the class weighting procedure described in Section 7.3 in order to determine the effect of class weighting on the performance of the model. Results on the English ACTER test dataset (heart failure) are presented in Table 17.

By applying the class weighting to reduce the effect of class imbalance, we manage to obtain substantial performance gains in terms of F1-score (see Table 17) for all tested models. The most significant improvements are demonstrated for the cased version of DistilBERT with the 156.51% and 195.07% increase on ANN and NES gold standards, respectively.

In Figure 1 we report the results of applying the term expansion and correction technique proposed in Section 7.3, using different IATE POS patterns. DistilBERT (uncased) model was used for these experiments. The patterns that had the best effect on the performance of the system were high-ratio IATE patterns including “ADJ NOUN”, “NOUN NOUN”, and “ADJ NOUN NOUN”, which can boost the original results from 5% points to more than 12% percentage points (when the “NOUN NOUN” pattern is used). Using the “NOUN ADJ NOUN” and “NOUN NOUN NOUN” patterns also leads to marginal F1-score improvements. Most IATE patterns longer than 3 words on the other hand drastically reduce the precision of the system, and this consequently also has a negative impact in terms of F1-score. This can be explained by the fact that rule-based expansion also introduces noise into the system since not all word sequences in the text with appropriate IATE POS patterns actually constitute a term. This means that the substantial improvements we get with the term expansion method in terms of recall always come at the expanse of the lower precision.

Table 17: Micro F1-scores obtained with and without applying class weights on the English ACTER dataset. The improvements are calculated using the Equation 1.

		F1-score		Improvement
Data	**Model**	**Non-weighted**	**Weighted**	**(%)**
ANN	BERT (uncased)	13.25	27.59	108.23
	BERT (cased)	15.96	30.13	88.79
	RoBERTa	14.73	**33.08**	124.58
	DistilBERT (uncased)	11.41	27.29	139.18
	DistilBERT (cased)	11.36	29.14	**156.51**
	BERT (multilingual-uncased)	15.41	29.18	89.35
	BERT (multilingual-cased)	14.77	21.80	47.59
	Average	13.84	28.32	107.75
NES	BERT (uncased)	16.54	36.44	120.31
	BERT (cased)	13.63	36.94	171.02
	RoBERTa	16.49	37.32	126.32
	DistilBERT (uncased)	13.59	37.74	177.70
	DistilBERT (cased)	12.77	37.68	**195.07**
	XLNet	17.80	**39.94**	124.38
	BERT (multilingual-uncased)	15.90	32.80	106.29
	BERT (multilingual-cased)	14.46	37.42	158.78
	Average	15.15	37.04	147.48

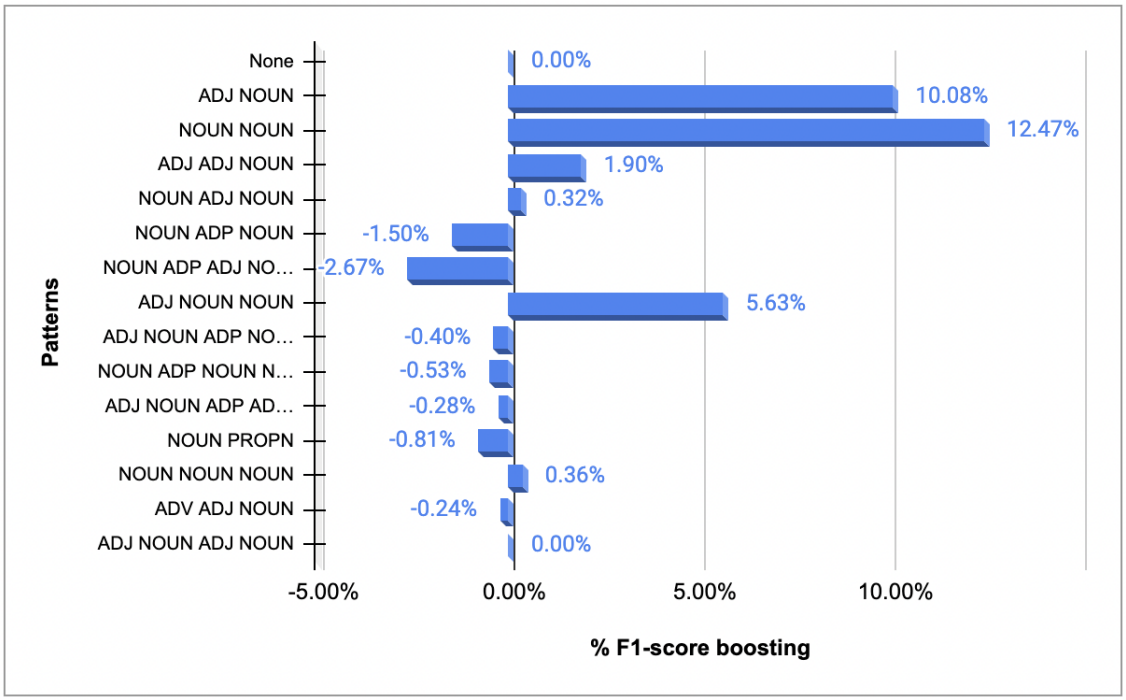


Figure 1: IATE pattern evaluation on terminology extraction performance.

Table 18: Evaluation of different models on the English ACTER dataset.

Models	ANN			NES		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
BERT (uncased)	45.18	19.86	27.59	49.67	28.78	36.44
BERT (cased)	43.52	23.04	30.13	46.32	30.72	36.94
RoBERTa	42.34	27.15	33.08	43.88	32.46	37.32
DistilBERT (uncased)	47.00	19.23	27.29	48.00	31.10	37.74
DistilBERT (cased)	46.31	21.26	29.14	46.89	31.49	37.68
XLNet	42.94	30.03	35.34	43.63	36.83	39.94
BERT (multilingual-uncased)	**47.97**	20.97	29.18	47.27	25.11	32.8
BERT (multilingual-cased)	44.99	14.44	21.86	**51.46**	29.4	37.42
TALN-LS2N	32.58	**72.68**	**44.99**	34.78	**0.87**	**46.66**
NYU	42.2	25.1	31.5	43.5	23.6	30.6

For the experiments on all of the languages in the ACTER dataset presented in the next section, we decided to apply only the class weighting procedure as it leads to improvements in terms of precision, recall, and overall F1-score, but not the term expansion approach given its influence on precision scores.

7.4.2 Multilingual evaluation

Results on the English ACTER test dataset (containg texts from the heart failure domain) for English cased and uncased BERT, multilingual BERT, RoBERTa, cased and uncased version of DistilBERT, and XLNet are presented in Table 18.

For French, we employ cased and uncased multilingual BERT, a multilingual cased DistilBERT and CamemBERT (Martin et al., 2019) (i.e., the French version of BERT), and report the results on the French ACTER test dataset (heart failure) in Table 19.

Lastly, we present the results of the multilingual version of DistilBERT, and cased and uncased versions of the multilingual BERT on the Dutch version of the heart failure test dataset in Table 20.

We compare our proposed approach to three baseline approaches by three teams that participated in the TermEval competition. Team NYU has applied an updated version of the Termolator (Meyers et al., 2018), the state-of-the-art rule-based approach towards term extraction. This baseline is only available for the English dataset. Similar to our approach, Team TALN-LS2N (Rigouts Terryn et al., 2020) used BERT, but rather than tackling the term extraction as a sequence labelling approach, they consider it a binary classification task. The model's input consists of the concatenation of a sentence and a selected n-gram within the sentence. If the n-gram is a term, the input is labelled as a positive training example. The inference in this approach is more time demanding than in ours since for each sentence all possible n-gram combinations in the sentence need to be tested in order for the model to determine, which of these combinations are in fact terms. This baseline was the winning method for term extraction in the TermEval competition and is available for English and French. Finally, for Dutch we compare our approach to the approach proposed by team NLPLab-UQAM (Rigouts Terryn et al., 2020), who fed pre-trained GloVe word embeddings to a bidirectional LSTM based neural architecture for term extraction. They achieved the best result in terms of F1-Score on the Dutch test set.

As can be seen, the performance of our approaches surpasses the baseline methods for French and Dutch in terms of F1-score. Our approach is nevertheless less competitive on the English test set, where the TALN-LS2N outperforms our approach in terms of F1-score by a large margin. However, our approach offers much higher precision than the baseline methods for all languages.

If we compare our best approach in terms of F1 on English (35.34% when XLNET model is used) to the NYU method (which is a pre-EMBEDDIA state-of-the-art method for term extraction), the proposed method offers an improvement of 12.19% in terms of F1-score.

Table 19: Evaluation of different models on the French ACTER dataset.

Models	ANN			NES		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
DistilBERT (multilingual-cased)	51.20	25.94	34.43	51.76	30.41	38.31
BERT (multilingual-uncased)	**54.02**	30.43	38.93	**55.09**	30.79	39.5
BERT (multilingual-cased)	52.04	32.05	39.67	53.1	30.62	38.84
CamemBERT	53.17	41.38	**46.54**	54.56	35.80	43.23
TALN-LS2N	41.88	**50.88**	45.94	45.17	51.55	**48.15**

For French, all tested models offer a much higher precision than the baseline. The best precision is achieved by the multilingual uncased BERT, who defeats the baseline on the ANN and NES gold standard with approximately 13% and 10% higher precision, respectively. Meanwhile, the French version of BERT – CamemBERT beats the baseline in terms of F1-score by a small margin (less than 1 percentage point).

Table 20: Evaluation of different models on the Dutch ACTER dataset.

Models	ANN			NES		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
DistilBERT (multilingual-cased)	58.81	51.16	54.72	59.91	56.70	58.26
BERT (multilingual-uncased)	**63.10**	59.21	**61.09**	**60.02**	**60.51**	**60.26**
BERT (multilingual-cased)	59.27	**59.35**	59.31	59.91	56.70	58.26
NLPLab-UQAM	18.10	19.30	18.60	18.90	18.60	18.70

The results on the Dutch dataset indicate that all the tested models demonstrate significantly higher F1-scores than the baseline NLPLab-UQAM approach, that is, results for all models are almost three times better than the baseline results on both ANN and NES gold standard lists. Unlike on the two other languages, here the recall of the proposed methods outperforms one of the baselines.

In conclusion, our method which tackles the terminology extraction as a sequence labelling task compares different transformer models and applies re-weighting represents a contribution to the state-of-the-art in the field. We plan to further include the final experiments using the pattern expansion and publish the paper on the topic.

8 Final evaluation of Term and Keywords Alignment (Task T2.2)

In this deliverable, we summarise the final evaluation of term matching approaches that were already briefly presented in D2.6, but since then resulted in two novel publications: (Repar & Shumakov, 2021) and (Repar et al., 2021).

8.1 Background

Bilingual terminology alignment is the process of aligning terms between two candidate term lists in two languages. The primary purpose of bilingual terminology extraction is to build a term bank - i.e. a list of terms in one language along with their equivalents in the other language.

The ability to accurately align concepts between languages can provide significant benefits in many practical applications. For example, in terminology, terms can be aligned between languages to provide bilingual terminological resources, while in the news industry, keywords can be aligned to provide better news clustering or search in another language. Accurate bilingual resources can also serve as seed data for various other NLP tasks, such as multilingual vector space alignment.

In previous deliverables D2.2 and D2.6, we already presented our approach (Repar, Martinc, & Pollak, 2019), which is based on the method by (Aker, Paramita, & Gaizauskas, 2013). In this deliverable, we present the final results on the application of this method on a media partners dataset, and the evaluation of this method by introducing embeddings-based features in the classification.

8.2 Methods

In the scope of the project, we first reimplemented the approach by (Aker et al., 2013) and proposed its adaptation. This was described in (Repar, Martinc, & Pollak, 2019) and Deliverables D2.3. Next, we applied this approach to tagset matching in a media setting and proposed additional features for the method, this was described in Deliverable D2.6 and in two papers published since then.

8.2.1 A machine-learning term alignment approach using a dictionary and cognate-based features

The approach that was described in (Repar, Martinc, & Pollak, 2019) and reported in deliverable D2.3. treats bilingual term alignment as a machine learning classification task. The approach is based on the study by (Aker et al., 2013) with several proposed adaptations. This work was the basis of term alignment evaluations in EMBEDDIA, and we briefly summarise it.

Considering term alignment as a bilingual classification task means that for each term pair, various features are created, and a classifier then assigns to each pair of terms a value saying if a term is a correct pair or not.

The approach considers a range of features of two types: dictionaries-based features are derived from Giza++ applied to the DGT translation memory, cognate-based features using the information on word similarities between languages and their combinations. The features are then used in an SVM classifier. For more details see (Repar, Martinc, & Pollak, 2019).

8.2.2 Application in media setting: tagset matching

We used the same approach as in (Repar, Martinc, & Pollak, 2019) for the Estonian-Russian language pair and then use the generated model to align tags provided by the Ekspress Meedia partner, where the task was that for every Russian keyword, we try to find an equivalent keyword in Estonian. The method was already introduced in D2.6, but we briefly summarise it and refer the reader to our paper (Repar & Shumakov, 2021) with the final description of the method.

In summary, as in (Repar, Martinc, & Pollak, 2019), the task was formulated as a machine learning classification problem, using a dictionary, cognate-based features, and their combinations as an input to an SVM classifier. For computing word similarity features, we had to apply an additional transliteration step to convert the scripts from Latin to Cyrillic. In addition, we had to identify a thesaurus with

aligned Russian-Estonian term pairs, and a parallel corpus for training the classifier. The resources are described in Section 8.3.

8.2.3 Term alignment with novel embeddings features

In addition in T2.2, we explored the potential of aligning embeddings instead of using dictionaries. The motivation is to reduce the reliance on large parallel corpora to derive dictionary-based features. The method was described in D2.6. Its final form appears in our publication accepted to Elex 2021 (Repar et al., 2021). We briefly summarise it here.

As a basis, we take the method by (Repar, Martinc, & Pollak, 2019). In order to generate additional features, we aligned monolingual fastText embeddings (Bojanowski, Grave, Joulin, & Mikolov, 2017b) using VecMap (Artetxe, Labaka, & Agirre, 2018) tool, which can align embeddings with the help of a small bilingual dictionary.

For alignment, we used a bilingual dictionary, compiled from two sources: single-word terms from Eurovoc and Wiktionary entries extracted using wikt2dict tool (Acs, 2014). From these aligned embedding vectors, we then calculated cosine distances between each Eurovoc term in one language and each Eurovoc term in the other language. For multi-word terms, we used the average (centroid) vector of all the words in a term.

Using the fastText-based lists of aligned words, we created 3-tuples[11] of most similar source-to-target and target-to-source words, such as:

- ksenofobija ['xenophobia', '0.744'], ['racism', '0.6797'], ['anti-semitism', '0.654']
- ženska ['woman', '0.7896'], ['women', '0.73'], ['female', '0.722']

The aligned words in the 3-tuple were sorted according to cosine similarity and these were then used to construct additional features for the machine learning algorithm.

The updated approach (that was tested on the English-Slovene pair) thus uses three types of features that express correspondences between the words (composing a term) in the target and source language. The dictionary and cognate-based features are the same as in (Repar, Martinc, & Pollak, 2019), while embeddings-based features are newly developed. The feature set consists of dictionary-based features (using Giza++), cognate-based features based on sting similarity, cognate-based features based on transliteration rules, combined dictionary and cognate-based features, term-length based features as well a s novel features derived from fastText embeddings alignments either used alone or in combination with cognate-based features.

The constructed features were then used to train an SVM binary classifier (Joachims, 2002) (with a linear kernel and the trade-off between training error and margin parameter c of 10). We selected three configurations from (Repar, Martinc, & Pollak, 2019) for comparison:

- Training set 1:200: a very unbalanced train set (ratio of 1:200 between positive and negative examples
- Training set filtering 3: In (Repar, Martinc, & Pollak, 2019), we have performed an error analysis and found that many incorrectly classified term pairs are cases of partial translation where one unit in a multi-word term has a correct Giza++ dictionary translation in the corresponding term in the other language. Based on this problem of partial translations, leading to false positive examples, we focused on the features that would eliminate this partial translations from the training set. After a systematic experimentation, we noticed that we can drastically improve precision if we only keep positive term pairs with the following feature values: isFirstWordTranslated = True, isLastWordTranslated = True, percentageOfCoverage > 0.66, isFirstWordTranslated-reversed = True, isLastWordTranslated-reversed = True, percentageOfCoverage-reversed > 0.66

[11] This number was determined experimentally.

- Cognates: the dataset is additionally filtered according to the following criteria: isFirstWordCognate = True and isLastWordCognate = True, isFirstWordTranslated = True and isLastWordCognate = True, isFirstWordCognate = True and isLastWordTranslated = True and we also use a Gaussian kernel instead of the linear one, since this new dataset structure represents a classic "exclusive or" (XOR) problem which a linear classifier is unable to solve.

For more details see the paper (Repar et al., 2021) in Appendix I.

8.3 Available resources

While benchmarking for the task of terminology alignment can theoretically use any terminological dictionary and domain corpus. A very suitable resource for term alignment in EU languages is EUROVOC thesaurus (Steinberger, Pouliquen, & Hagman, 2002) with terminology in several EU languages, and using European translation resources, such as Europarl (Koehn, 2005) or DG translation memory (Steinberger, Eisele, Klocek, Pilos, & Schlüter, 2013) as the corpora for deriving representations.

For media setting keyword alignment (Repar & Shumakov, 2021), we used the dataset of Estonian and Russian tags that was provided by Ekspress Meedia as a simple list of one tag per line. The tagset consists of keywords that journalists assigned to articles to describe an article's topic. The total number of tags was 65,830, and Russian and Estonian tags were provided in random order. Since Russian and Estonian use different writing scripts (Cyrillic vs Latin), we were able to separate the tags using a simple regular expression to detect Cyrillic characters. The number of Russian tags was 6,198 and they were mixed with the Estonian tags in random order. The vast majority of the tags are either unigrams or bigrams. However, one should note that these tags were not aligned, and in our experiments, a manual evaluation of alignment quality was performed.

As shown in our experiments, for deriving the representations or dictionaries for non-EU languages, one cannot rely on the DGT translation memory and the Eurovoc thesaurus support. For the parallel corpus, available resources can be found in Opus portal[12]. We tested the Estonian Open Parallel corpus[13] and the Estonian-Russian OpenSubtitles corpus. The OpenSubtitles corpus performed better, most likely due to its much larger size (85,449 parallel Estonian-Russian segments in the Estonian Open Parallel corpus vs. 7.1 million segments in the OpenSubtitles corpus).

While finding parallel Estonian-Russian corpora was trivial due to the list of available corpora on the Opus portal, finding an appropriate bilingual terminological database proved to be more difficult. Ideally, we would want to use a media or news-related Estonian-Russian terminological resource, but to the best of our knowledge, there was none available. Note that the terminological resource needs to have at least several thousand entries: the Eurovoc version used by (Repar, Martinc, & Pollak, 2019) contained 7,083 English-Slovene term pairs. We finally settled on the environmental thesaurus Gemet[14], which at the time had 3,721 Estonian-Russian term pairs.

As shown in (Repar et al., 2021), one can replace the need for large parallel corpora by using alignment of monolingual embedding models, where for building seed dictionaries wikt2dict (Acs, 2014) is an appropriate solution.

8.4 Results

We present in the following sections the results obtained regarding tagset and term alignment.

[12] https://opus.nlpl.eu/
[13] https://doi.org/10.15155/9-00-0000-0000-0000-0002AL
[14] https://www.eionet.europa.eu/gemet/en/themes/

Table 21: Results on the English-Slovenian term pair.

No.	Config EN-SL	Training set size	Pos/Neg ratio	Precision	Recall	F1-score
Dictionary-based and cognate-based features						
1	Training set 1:200	1,303,083	1:200	0.4299	**0.7617**	0.5496
2	Training set filtering 3	645,813	1:200	0.9342	0.4966	0.6485
3	Cognates approach	672,345	1:200	0.8732	0.5167	0.6492
Dictionary-based, embedding-based and cognate-based features						
1	Training set 1:200	1,303,083	1:200	0.5375	0.680	0.6004
2	Training set filtering 3	695,058	1:200	0.8170	0.5133	0.6305
3	Cognates approach	706,113	1:200	0.8991	0.5200	**0.6589**
Embedding-based and cognate-based features only						
1	Training set 1:200	1,303,083	1:200	0.3232	0.4967	0.3916
2	Training set filtering 3	322,605	1:200	0.9545	0.2450	0.3899
3	Cognates approach	394,362	1:200	**0.9618**	0.3617	0.5242

8.4.1 ExM keyword tagset alignment

Since the ExM dataset was not aligned, we were unable to calculate precision, recall, and f-score and we instead conducted a manual evaluation of aligned pairs by a domain expert. The alignment resulted in 4,989 positively classified Estonian-Russian tag pairs. A subset of these (500) was manually evaluated by a person with knowledge of both languages provided by Ekspress Meedia according to the following methodology:

- C: if the tag pair is a complete match
- P: if the tag pair is a partial match, i.e. when a multiword tag in one language is paired with a single word tag in the other language (e.g. eesti kontsert — концерт , or *Estonian concert — concert*)
- N: if the tag pair is a no match

Of the 500 positively classified tag pairs that were manually evaluated, 49% percent were deemed to be complete matches, a further 25% were evaluated as partial matches, and 26% were considered to be wrongly classified as positive tag pairs. For detailed evaluation, see (Repar & Shumakov, 2021).

8.4.2 Term alignment with novel embeddings features

We performed two sets of experiments (described already in D2.6, but now finalised and published (Repar et al., 2021): first, we simply added the new embedding-based features to the dataset, and then we remove the dictionary-based features from the dataset to see whether the novel embedding-based features could replace them without a major impact to the performance. As can be observed from Table 21, the results are a mixed bag when using all available features. Without any training set filtering, the new features improve precision at the expense of recall but are less effective when filtering is applied. Nevertheless, when we use additional train set filters for the cognates approach, we can observe a slight increase in both precision and recall resulting in the overall highest F-score. When we use only embedding-based and cognate-based features, there is a significant drop in recall in all cases, but precision actually increases when train set filtering is applied and the Cognates approach achieves the overall best precision.

9 Conclusions and Future Work

This deliverable is the *Final evaluation report on advanced cross-lingual NLP technology* of WP2, *Advanced NLP Technologies for Less-Resourced Languages*, presenting the outcomes related to task T2.4, *Public resource gathering, benchmarking and evaluation*.

Specifically, this deliverable focuses on six different NLP tasks, that are grouped in two WP2 tasks, T2.1 and T2.2. For each of these NLP tasks, we included a list of available resources, our last developed methods, and a comparison with respect to the state of the art.

With respect to T2.1, Cross-lingual Semantic Enrichment, Named Entity Recognition was the easiest NLP task to realise. On average, we achieved an F1-score of 0.942, for all EMBEDDIA languages. This contrasted with the state of the art at the beginning of the project, where the average was 0.867. The improvement was achieved thanks to the use of multiple BERT-based language models, which were fine-tuned. As well, the technologies developed in EMBEDDIA were used in the 2021 SlavNER challenge, for creating a NER system capable of predicting named entities in Slavic languages. Our participating team achieved the 2nd place at the competition.

Regarding Named Entity Linking, we managed to improve on average 38% the F1-score with respect to the state of the art at the beginning of EMBEDDIA. This was achieved thanks to the use of a multilingual approach and training the models over multiples languages, instead of using only English.

As well, part of the WP2 team participated in the annotation of data for SlavNER 2021. This resulted in a new tagged dataset that covers multiple Slavic languages, two of which are used in EMBEDDIA. The annotation is regarding named entities and named entity linking.

We observed that Event Detection continues to be challenging. However, the models produced in EMBEDDIA managed to increase the F1-score by at least 39%. Furthermore, in some languages, such as Czech and Russian, we managed to pass from an F1-score of 0 to an F1-score of at least 0.680. The increment in the performance was obtained thanks to transfer learning from larger pre-trained language models.

Concerning T2.2, we show that the keyword extraction models developed during the EMBEDDIA project significantly outperform pre-EMBEDDIA state-of-art keyword extraction methods on all datasets, for which comparison is available. On average, we achieve an improvement of 38.93% in terms of F1@10 and improvement of 26.04% in terms F1@5. We also show that we can obtain large gains in the recall of neural keyword extraction models by combining them with the TF-IDF(tm) keyword extraction method. Finally, besides developing new state-of-the-art models, the T2.2 team collaborated in creation of keyword extraction datasets based on data from the EMBEDDIA media partners. We have also shown that similar to out keyword extraction experiments, same sequence labelling setting can be applied to the term extraction task, and achieves competitive performance on gold standard datasets. I addition term alignment experiments were finalised.

As future work, we will continue applying the technologies developed in WP2 to the tasks in WP3 and WP4, and additionally test selected models (keywords) with the media partners. As well, we will pursue new publications in order to expand the relevance of EMBEDDIA in the NLP field.

10 Associated Outputs

The work described in this deliverable has resulted in the following software resources:

Description	URL	Availability
Event Detection	https://github.com/EMBEDDIA/event-detection	To become public
NER BERT Multi-task	https://github.com/EMBEDDIA/NER_BERT_Multitask	Public (MIT)
NER FEDA	https://github.com/EMBEDDIA/NER_FEDA	Public (MIT)
NEL Filter	https://github.com/EMBEDDIA/NEL_Filter	Public (MIT)
Stacked NER	https://github.com/EMBEDDIA/stacked-ner	Public (MIT)

Works marked as *To become public* mean that they are available only within the consortium while the associated work is yet to be published. They will be released publicly when the associated work is published.

We present in Table 22 the publications that have been produced between December 2020 and June 2021 and that are related to this deliverable.

Table 22: Publications related to this deliverable.

Citation	Status	Appendix
Cabrera-Diego, L. A., Moreno, J. G. and Doucet, A. Using a Frustratingly Easy Domain and Tagset Adaptation for Creating Slavic Named Entity Recognition Systems. Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2021)	Published	A
Boroş, E., Hamdi, A., Linhares Pontes, E., Cabrera-Diego, L.A, Moreno, J. G., Sidere, N., Doucet, A. Atténuer les erreurs de numérisation dans la reconnaissance d'entités nommées pour les documents historiques. Actes du 17ème édition de CORIA	Published	B
Boroş, E., Besançon, R., Ferret, O., Grau, B. Intérêt des modèles de caractères pour la détection d'événements. (Accepted at TALN 2021 (Conf.))	Accepted	C
Linhares Pontes, E., Cabrera-Diego, L.A, Moreno, J. G., Boroş, E., Hamdi, A., Doucet, A, Sidere, N., Coustaty, M.. MELHISSA: A Multilingual Entity Linking Architecture for Historical Press Articles (To be submitted to IJDL (Journal))	To be submitted	D
Thi, H. H. T., Doucet, A., Sidere, N., Moreno, J. G., Pollak, S. Named entity recognition architecture combining contextual and global features (To submit to ICADL (Conference))	To be submitted	E
Piskorski, J., Babych, B., Kancheva, Z., Kanishcheva, O., Lebedeva, M., Marcińczuk, M., Nakov, P., Osenova, P., Pivovarova, L., Pollak, S., et al. Slav-NER: the 3rd Cross-lingual Challenge on Recognition, Normalization, Classification, and Linking of Named Entities across Slavic Languages. Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2021)	Published	F
Koloski, B., Pollak, S., Škrlj, B., Martinc, M. Extending Neural Keyword Extraction with TF-IDF tagset matching. Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation	Published	G
Repar, A., Shumakov, A. Aligning Estonian and Russian news industry keywords with the help of subtitle translations and an environmental thesaurus. Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation	Published	H
Repar, A., Martinc, M., Ulčar, M., Pollak, S. Word-embedding based bilingual terminology alignment. Accepted at eLex 2021 (Conference)	Accepted	I

References

Acs, J. (2014). Pivot-based multilingual dictionary building using wiktionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation LREC.*

Aker, A., Paramita, M., & Gaizauskas, R. (2013). Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vol. 1, pp. 402–411).

Arkhipov, M., Trofimova, M., Kuratov, Y., & Sorokin, A. (2019, August). Tuning Multilingual Transformers for Language-Specific Named Entity Recognition. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing* (pp. 89–93). Florence, Italy: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W19-3712 doi: 10.18653/v1/W19-3712

Artetxe, M., Labaka, G., & Agirre, E. (2018). Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-Second AAAI Conference on Artificial Intelligence.*

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017a). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, 135–146. Retrieved from https://www.aclweb.org/anthology/Q17-1010 doi: 10.1162/tacl_a_00051

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017b). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, 135–146. Retrieved from https://www.aclweb.org/anthology/Q17-1010 doi: 10.1162/tacl_a_00051

Boros, E., Hamdi, A., Linhares Pontes, E., Cabrera-Diego, L. A., Moreno, J. G., Sidere, N., & Doucet, A. (2020, November). Alleviating Digitization Errors in Named Entity Recognition for Historical Documents. In *Proceedings of the 24th Conference on Computational Natural Language Learning* (pp. 431–441). Online: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2020.conll-1.35

Boros, E., Linhares Pontes, E., Cabrera-Diego, L. A., Hamdi, A., Moreno, J. G., Sidere, N., & Doucet, A. (2020). Robust Named Entity Recognition and Linking on Historical Multilingual Documents. In L. Cappellato, C. Eickhoff, N. Ferro, & A. Névéol (Eds.), *CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum.* CEUR-WS.

Boros, E., Moreno, J. G., & Doucet, A. (2021a). Event detection as question answering with entity information. *arXiv preprint arXiv:2104.06969*.

Boros, E., Moreno, J. G., & Doucet, A. (2021b). Event detection with entity markers. In D. Hiemstra, M. Moens, J. Mothe, R. Perego, M. Potthast, & F. Sebastiani (Eds.), *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part II* (Vol. 12657, pp. 233–240). Springer. Retrieved from https://doi.org/10.1007/978-3-030-72240-1_20 doi: 10.1007/978-3-030-72240-1_20

Bougouin, A., Boudin, F., & Daille, B. (2013). TopicRank: Graph-based topic ranking for keyphrase extraction. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)* (pp. 543–551).

Broscheit, S. (2020). Investigating entity knowledge in bert with simple neural end-to-end entity linking. *arXiv preprint arXiv:2003.05473*.

Cabrera-Diego, L. A., Moreno, J. G., & Doucet, A. (2021a). Simple ways to improve NER in every language using markup. In Elena Demidova, Sherzod Hakimov, Jane Winters, & Marko Tadić (Eds.), *Proceedings of the 2nd International Workshop on Cross-lingual Event-centric Open Analytics (CLEOPATRA 2021)* (Vol. 2829, pp. 17–31). Ljubljana, Slovenia: CEUR-WS. Retrieved from http://ceur-ws.org/Vol-2829/paper2.pdf

Cabrera-Diego, L. A., Moreno, J. G., & Doucet, A. (2021b, April). Using a Frustratingly Easy Domain and Tagset Adaptation for Creating Slavic Named Entity Recognition Systems. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing* (pp. 98–104). Kiyv, Ukraine: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2021.bsnlp-1.12

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., & Jatowt, A. (2018). Yake! collection-independent automatic keyword extractor. In *European Conference on Information Retrieval* (pp. 806–810).

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., ... Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., & Jégou, H. (2017). Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Cucerzan, S. (2007, June). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (pp. 708–716). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/D07-1074

Daille, B., Gaussier, É., & Langé, J.-M. (1994). Towards automatic extraction of monolingual and bilingual terminology. In *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics.*

Dalianis, H., & Åström, E. (2001). *SweNam-A Swedish Named Entity recognizer Its construction, training and evaluation* (Tech. Rep. No. TRITA-NA-P0113 - IPLab-189.) Stockholm, Sweden: Department of Numerical Analysis and Computing Science, KTH Royal Institute of Technology.

Daumé III, H. (2007, June). Frustratingly Easy Domain Adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (pp. 256–263). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/P07-1033

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (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)* (pp. 4171–4186). Association for Computational Linguistics. doi: 10.18653/v1/N19-1423

El-Beltagy, S. R., & Rafea, A. (2009). Kp-miner: A keyphrase extraction system for english and arabic documents. *Information Systems*, *34*(1), 132–144.

Gallina, Y., Boudin, F., & Daille, B. (2019). Kptimes: A large-scale dataset for keyphrase generation on news documents. *arXiv preprint arXiv:1911.12559*.

Ganea, O.-E., & Hofmann, T. (2017). Deep joint entity disambiguation with local neural attention. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2619–2629). Association for Computational Linguistics. Retrieved from http://aclweb.org/anthology/D17-1277 doi: 10.18653/v1/D17-1277

Grishman, R., & Sundheim, B. (1996). Message Understanding Conference-6: A Brief History. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics.* Retrieved from https://www.aclweb.org/anthology/C96-1079

Hazem, A., Bouhandi, M., Boudin, F., & Daille, B. (2020). Termeval 2020: Taln-ls2n system for automatic term extraction. In *Proceedings of the 6th International Workshop on Computational Terminology* (pp. 95–100).

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

Justeson, J. S., & Katz, S. M. (1995). Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural language engineering*, *1*(1), 9–27.

Kageura, K., & Umino, B. (1996). Methods of automatic term recognition: A review. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, *3*(2), 259–289.

Kapočiūtė, J., & Raškinis, G. (2005). Rule-based Annotation of Lithuanian Text Corpora. *Information Technology and Control*, *34*(3), 290–296.

Kim, Y.-B., Stratos, K., & Sarikaya, R. (2016, December). Frustratingly Easy Neural Domain Adaptation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (pp. 387–396). Osaka, Japan: The COLING 2016 Organizing Committee. Retrieved from https://www.aclweb.org/anthology/C16-1038

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit* (Vol. 5, pp. 79–86).

Kolitsas, N., Ganea, O.-E., & Hofmann, T. (2018). End-to-end neural entity linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning* (pp. 519–529). Association for Computational Linguistics. Retrieved from http://aclweb.org/anthology/K18-1050

Koloski, B., Pollak, S., Škrlj, B., & Martinc, M. (2021, April). Extending neural keyword extraction with TF-IDF tagset matching. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation* (pp. 22–29). Online: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2021.hackashop-1.4

Koloski, B., Pollak, S., Škrlj, B., & Martinc, M. (2021). *Keyword extraction datasets for Croatian, Estonian, Latvian and Russian 1.0*. Retrieved from http://hdl.handle.net/11356/1403 (Slovenian language resource repository CLARIN.SI)

Kuratov, Y., & Arkhipov, M. (2019). *Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language*. (_eprint: 1905.07213)

Lejeune, G., Brixtel, R., Doucet, A., & Lucas, N. (2015). Multilingual event extraction for epidemic detection. *Artificial intelligence in medicine*, *65*(2), 131–143.

Li, J., Sun, A., Han, J., & Li, C. (2020). A Survey on Deep Learning for Named Entity Recognition. *IEEE Transactions on Knowledge and Data Engineering*, 1–1. doi: 10.1109/TKDE.2020.2981314

Li, J., Sun, Y., Johnson, R. J., Sciaky, D., Wei, C.-H., Leaman, R., ... Lu, Z. (2016, May). BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database : the journal of biological databases and curation*, *2016*, baw068. Retrieved from https://pubmed.ncbi.nlm.nih.gov/27161011 (Publisher: Oxford University Press) doi: 10.1093/database/baw068

Linhares Pontes, E., Cabrera-Diego, L. A., Moreno, J. G., Boros, E., Hamdi, A., Sidère, N., ... Doucet, A. (2020). Entity Linking for Historical Documents: Challenges and Solutions. In E. Ishita, N. L. S. Pang, & L. Zhou (Eds.), *Proceedings of the 22nd International Conference on Asia-Pacific Digital Libraries (ICADL 2020)* (pp. 215–231). Kyoto, Japan: Springer International Publishing. doi: 10.1007/978-3-030-64452-9_19

Linhares Pontes, E., Doucet, A., & Moreno, J. G. (2020). Linking named entities across languages using multilingual word embeddings. In *Jointed Conference on Digital Libraries (JCDL) 2020.*

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ljubešić, N., & Lauc, D. (2021, April). BERTić - the transformer language model for Bosnian, Croatian, Montenegrin and Serbian. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*

(pp. 37–42). Kiyv, Ukraine: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2021.bsnlp-1.5

Luoma, J., Oinonen, M., Pyykönen, M., Laippala, V., & Pyysalo, S. (2020, May). A Broad-coverage Corpus for Finnish Named Entity Recognition. In *Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 4615–4624). Marseille, France: European Language Resources Association. Retrieved from https://www.aclweb.org/anthology/2020.lrec-1.567

Malmsten, M., Börjeson, L., & Haffenden, C. (2020). Playing with Words at the National Library of Sweden – Making a Swedish BERT. *arXiv cs.CL*. (_eprint: 2007.01658)

Martin, L., Muller, B., Suárez, P. J. O., Dupont, Y., Romary, L., de la Clergerie, É. V., ... Sagot, B. (2019). Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

Martinc, M., Škrlj, B., & Pollak, S. (2021). Tnt-kid: Transformer-based neural tagger for keyword identification. *Natural Language Engineering*, 1–40. doi: 10.1017/S1351324921000127

Meng, R., Zhao, S., Han, S., He, D., Brusilovsky, P., & Chi, Y. (2017). Deep keyphrase generation. *arXiv preprint arXiv:1704.06879*.

Meyers, A. L., He, Y., Glass, Z., Ortega, J., Liao, S., Grieve-Smith, A., ... Babko-Malaya, O. (2018). The termolator: terminology recognition based on chunking, statistical and search-based scores. *Frontiers in Research Metrics and Analytics*, *3*, 19.

Mihalcea, R., & Tarau, P. (2004). Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing* (pp. 404–411).

Moreno, J. G., Linhares Pontes, E., Coustaty, M., & Doucet, A. (2019, August). TLR at BSNLP2019: A Multilingual Named Entity Recognition System. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing* (pp. 83–88). Florence, Italy: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/W19-3711 doi: 10.18653/v1/W19-3711

Munnelly, G., & Lawless, S. (2018). Investigating entity linking in early english legal documents. In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries* (p. 59–68). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3197026.3197055

Mutuvi, S., Boros, E., Doucet, A., Lejeune, G., Jatowt, A., & Odeo, M. (2020). Multilingual epidemiological text classification: A comparative study. In *COLING, International Conference on Computational Linguistics.*

Onoe, Y., & Durrett, G. (2020). Fine-grained entity typing for domain independent entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 34, pp. 8576–8583).

Piskorski, J., Babych, B., Kancheva, Z., Kanishcheva, O., Lebedeva, M., Marcińczuk, M., ... Yangarber, R. (2021, April). Slav-NER: the 3rd cross-lingual challenge on recognition, normalization, classification, and linking of named entities across Slavic languages. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing* (pp. 122–133). Kiyv, Ukraine: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2021.bsnlp-1.15

Pollak, S., Robnik Šikonja, M., Purver, M., Boggia, M., Shekhar, R., Pranjić, M., ... Doucet, A. (2021, April). EMBEDDIA tools, datasets and challenges: Resources and hackathon contributions. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation.* Association for Computational Linguistics.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, *1*(8).

Raiman, J., & Raiman, O. (2018, Apr.). Deeptype: Multilingual entity linking by neural type system evolution. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). Retrieved from https://ojs.aaai.org/index.php/AAAI/article/view/12008

Reimers, N., & Gurevych, I. (2019). *Alternative Weighting Schemes for ELMo Embeddings*. (_eprint: 1904.02954)

Repar, A., Martinc, M., & Pollak, S. (2019, Nov). Reproduction, replication, analysis and adaptation of a term alignment approach. *Language Resources and Evaluation*. Retrieved from `https://doi.org/10.1007/s10579-019-09477-1` doi: 10.1007/s10579-019-09477-1

Repar, A., Martinc, M., Ulčar, M., & Pollak, S. (2021). Word-embedding based bilingual terminology alignment. In *Electronic lexicography in the 21st century. Proceedings of the eLex 2021 conference.*

Repar, A., Podpečan, V., Vavpetič, A., Lavrač, N., & Pollak, S. (2019). TermEnsembler: An ensemble learning approach to bilingual term extraction and alignment. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, *25*(1), 93–120.

Repar, A., & Shumakov, A. (2021, April). Aligning Estonian and Russian news industry keywords with the help of subtitle translations and an environmental thesaurus. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation* (pp. 71–75). Online: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/2021.hackashop-1.10`

Rigouts Terryn, A., Hoste, V., Drouin, P., & Lefever, E. (2020). Termeval 2020: Shared task on automatic term extraction using the annotated corpora for term extraction research (acter) dataset. In *6th International Workshop on Computational Terminology (COMPUTERM 2020)* (pp. 85–94).

Rigouts Terryn, A., Hoste, V., & Lefever, E. (2019, Mar 26). In no uncertain terms: a dataset for monolingual and multilingual automatic term extraction from comparable corpora. *Language Resources and Evaluation*. Retrieved from `https://doi.org/10.1007/s10579-019-09453-9` doi: 10.1007/s10579-019-09453-9

Rijhwani, S., Xie, J., Neubig, G., & Carbonell, J. (2019, January). Zero-shot neural transfer for cross-lingual entity linking. In *Thirty-Third AAAI Conference on Artificial Intelligence (AAAI).* Honolulu, Hawaii. doi: 10.1609/aaai.v33i01.33016924

Rose, S., Engel, D., Cramer, N., & Cowley, W. (2010). Automatic keyword extraction from individual documents. *Text mining: applications and theory*, 1–20.

Ruiz, P., & Poibeau, T. (2019, March). Mapping the Bentham Corpus: Concept-based Navigation. *Journal of Data Mining and Digital Humanities.*, *Special Issue: Digital Humanities between knowledge and know-how (Atelier Digit_Hum)*. Retrieved from `https://hal.archives-ouvertes.fr/hal-01915730`

Sahrawat, D., Mahata, D., Kulkarni, M., Zhang, H., Gosangi, R., Stent, A., ... Zimmermann, R. (2020). Keyphrase extraction from scholarly articles as sequence labeling using contextualized embeddings. In *Proceedings of European Conference on Information Retrieval (ECIR 2020)* (pp. 328–335).

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Shen, W., Wang, J., & Han, J. (2014). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, *27*(2), 443–460.

Škrlj, B., Repar, A., & Pollak, S. (2019). RaKUn: Rank-based keyword extraction via unsupervised learning and meta vertex aggregation. In *International Conference on Statistical Language and Speech Processing* (pp. 311–323).

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., & Schlüter, P. (2013). DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC'2012).*

Steinberger, R., Pouliquen, B., & Hagman, J. (2002). Cross-lingual document similarity calculation using the multilingual thesaurus eurovoc. *Computational Linguistics and Intelligent Text Processing*, 101–121.

Sterckx, L., Demeester, T., Deleu, J., & Develder, C. (2015). Topical word importance for fast keyphrase extraction. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 121–122).

Sun, S., Xiong, C., Liu, Z., Liu, Z., & Bao, J. (2020). Joint keyphrase chunking and salience ranking with bert. *arXiv preprint arXiv:2004.13639*.

Tanvir, H., Kittask, C., & Sirts, K. (2020). *EstBERT: A Pretrained Language-Specific BERT for Estonian*. (_eprint: 2011.04784)

Tjong Kim Sang, E. F. (2002). Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002).* Retrieved from `https://www.aclweb.org/anthology/W02-2024`

Tjong Kim Sang, E. F., & De Meulder, F. (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* (pp. 142–147). Retrieved from `https://www.aclweb.org/anthology/W03-0419`

Toivonen, H., & Boggia, M. (Eds.). (2021, April). *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation.* Online: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/2021.hackashop-1.0`

Tsygankova, T., Mayhew, S., & Roth, D. (2019, August). BSNLP2019 Shared Task Submission: Multisource Neural NER Transfer. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing* (pp. 75–82). Florence, Italy: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/W19-3710` doi: 10.18653/v1/W19-3710

Ulčar, M., & Robnik-Šikonja, M. (2020). FinEst BERT and CroSloEngual BERT. In P. Sojka, I. Kopeček, K. Pala, & A. Horák (Eds.), *Text, Speech, and Dialogue* (pp. 104–111). Cham: Springer International Publishing.

Valmarska, A., Cabrera-Diego, L., Linhares Pontes, E., & Pollak, S. (2021, April). Exploratory Analysis of News Sentiment Using Subgroup Discovery. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing* (pp. 66–72). Kiyv, Ukraine: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/2021.bsnlp-1.7`

van Hooland, S., De Wilde, M., Verborgh, R., Steiner, T., & Van de Walle, R. (2013, 11). Exploring entity recognition and disambiguation for cultural heritage collections. *Digital Scholarship in the Humanities*, *30*(2), 262-279. doi: 10.1093/llc/fqt067

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Vintar, Š. (2010). Bilingual term recognition revisited: The bag-of-equivalents term alignment approach and its evaluation. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, *16*(2), 141–158.

Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., ... Pyysalo, S. (2019). *Multilingual is not enough: BERT for Finnish*. (_eprint: 1912.07076)

Wan, X., & Xiao, J. (2008). Single document keyphrase extraction using neighborhood knowledge. In *Proceedings of the AAAI Conference* (Vol. 8, pp. 855–860).

Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., & Nevill-Manning, C. G. (2005). Kea: Practical automated keyphrase extraction. In *Design and Usability of Digital Libraries: Case Studies in the Asia Pacific* (pp. 129–152). IGI Global.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Zhou, S., Rijhwani, S., & Neubig, G. (2019, November). Towards zero-resource cross-lingual entity linking. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)* (pp. 243–252). China: ACL.

Znotiņš, A., & Guntis Barzdiņš. (2020). LVBERT: Transformer-Based Model for Latvian Language Understanding. In Andrius Utka, Jurgita Vaičenonienė, Jolanta Kovalevskaitė, & Danguolė Kalinauskaitė (Eds.), *Human Language Technologies – The Baltic Perspective: Proceedings of the Ninth International Conference Baltic HLT 2020* (Vol. 328, pp. 111 – 115). Kaunas, Lithuania: IOS Press. doi: 10.3233/FAIA200610

Appendices

A Using a Frustratingly Easy Domain and Tagset Adaptation for Creating Slavic Named Entity Recognition Systems

Using a Frustratingly Easy Domain and Tagset Adaptation for Creating Slavic Named Entity Recognition Systems

Luis Adrián Cabrera-Diego
La Rochelle Université, L3i,
La Rochelle, 17031, France
luis.cabrera_diego@univ-lr.fr

Jose G. Moreno
Université Paul Sabatier, IRIT,
Toulouse, 31062, France
jose.moreno@irit.fr

Antoine Doucet
La Rochelle Université, L3i,
La Rochelle, 17031, France
antoine.doucet@univ-lr.fr

Abstract

We present a collection of Named Entity Recognition (NER) systems for six Slavic languages: Bulgarian, Czech, Polish, Slovenian, Russian and Ukrainian. These NER systems have been trained using different BERT models and a Frustratingly Easy Domain Adaptation (FEDA). FEDA allow us creating NER systems using multiple datasets without having to worry about whether the tagset (e.g. Location, Event, Miscellaneous, Time) in the source and target domains match, while increasing the amount of data available for training. Moreover, we boosted the prediction on named entities by marking uppercase words and predicting masked words. Participating in the 3rd Shared Task on SlavNER[1], our NER systems reached a strict micro F-score of up to 0.908. The results demonstrate good generalization, even in named entities with weak regularity, such as book titles, or entities that were never seen during the training.

1 Introduction

Named Entity Recognition (NER) is a fundamental task in domain of Natural Language Processing (NLP) that consists of extracting entities that semantically refer to aspects such as locations, people or organizations (Luoma et al., 2020). Since the creation of BERT (Devlin et al., 2019), multiple NER systems have brought the state of the art to new levels of performance. Nonetheless, there are many challenges that still need to be faced, especially in the case of less-resources languages.

In the *2nd Shared Task on SlavNER* (Piskorski et al., 2019), the top-two systems in the detection Named Entities (NEs), Tsygankova et al. (2019) and Arkhipov et al. (2019), managed to reach a relaxed partial micro F-score of 0.9, followed by two other systems with values slightly better than 0.8 (Moreno et al., 2019). For the *3rd Shared Task on SlavNER*, we consider that in order to improve the scores, in terms of the strict evaluation, and NEs related to products and events, it is necessary to include additional data that could improve the generalization of the models to any kind of topic.

While in the literature there are multiple techniques for training models over additional datasets, such as transfer learning and domain adaptation, using these techniques might pose additional questions. For example, to determine which layers to freeze, fine-tune or substitute. Furthermore, different datasets might use dissimilar tagsets, which might be incompatible (Nozza et al., 2021).

In this paper, we present the participation of laboratory *L3i* in the 3rd Shared Task on SlavNER. Specifically, we participate with multiple NER systems for Slavic languages using different BERT models and training over diverse datasets through a *Frustratingly Easy Domain Adaptation* (FEDA) algorithm (Daumé III, 2007; Kim et al., 2016).[2] The FEDA algorithm has for objective to learn common and domain-specific patterns between multiple datasets, while keeping separately patterns belonging only to the domain-specific data (Daumé III, 2007). Particularly, the use of FEDA allow us sharing the knowledge and patterns found in multiple datasets without having to worry about which different tagsets are used among them.

Apart from the FEDA algorithm, we explore some other techniques that might improve the performance of our NER system based on the ideas of Cabrera-Diego et al. (2021). Specifically, we analyze whether the marking and enrichment of uppercase tokens can improve the detection of NEs. As well, we use the prediction of masked tokens as a way to improve NER systems' generalization.

[1] bsnlp.cs.helsinki.fi/shared-task.html, last visited on 9 March 2021

[2] github.com/EMBEDDIA/NER_FEDA

Proceedings of the 8th BSNLP Workshop on Balto-Slavic Natural Language Processing, pages 98–104

The rest of the paper is organized as follows. In Section 2, we introduce the background for the proposed work. This is followed by the methodology in Section 3. The data and the experimental settings are described in Section 4 and Section 5, respectively. In Section 6, we present the results obtained. Finally, the conclusions and future work are detailed in Section 7.

2 Background

Uppercase sentences: Although most of the NER corpora found in the literature provide texts following standard case rules, it is not infrequent to find datasets containing some sentences in which all the words are in uppercase, e.g. English CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003) or SSJ500k (Krek et al., 2019). In NLP systems based on BERT or similar, where Byte Pair Encoding (BPE) tokenizers are used, the presence of uppercase sentences might pose a greater challenge than standard case sentences. The reason is that an uppercase word have different BPE tokens with respect to its lower and title-case versions, and in consequence different dense representation (Powalski and Stanislawek, 2020; Sun et al., 2020).

Weak generalization: One of the most challenging aspects of NER systems is to deal with NEs that have a weak or zero regularity, such as names of movies, and NEs that were never seen during training (Lin et al., 2020b). Some methods found in the literature for improving generalization consists of learning manually defined triggers (Lin et al., 2020a), but also permuting NEs and reducing context such as in Lin et al. (2020b).

FEDA: Originally proposed by Daumé III (2007), the FEDA was firstly designed for sparse machine learning algorithms. Later, Kim et al. (2016), proposed a neural network version of this domain adaptation algorithm. While the former resides in duplicating input features, the latter consists of activating specific neural network layers.

3 Methodology

Consider $\mathcal{D} = \{D_1, D_2, \ldots, D_n | n > 1\}$ a collection of datasets from which we want to train a model. Furthermore, consider a classifier C a stack of two linear layers in which in between we set an activation layer ReLU and a dropout. The first linear layer has a size of 512, while the output h produced by C has a size of l, which is the number of different labels found in $\mathcal{D}$. Thus, the proposed model for doing the FEDA consists of adding on top of BERT $n + 1$ classifiers such that we have $\mathcal{C} = \{C_0, C_1, C_2, \ldots, C_n\}$. The classifier C_0 represents a general classifier that will receive as input the sentences from all the datasets in $\mathcal{D}$, while $C_k \in \{\mathcal{C} | 0 < k \leq n\}$ represent a specialized classifier that will focus only on the sentences that belong to the dataset $D_k \in \{\mathcal{D} | 0 < k \leq n\}$. For each sentence belonging to a dataset D_k, we do the element-wise sum between h_0 and h_k, i.e. $H_k = h_0 + h_k$. Finally, H_k is introduced it into a CRF layer, which will determine the labels of each word in a sentence. Figure 1 depicts the proposed architecture.

For increasing the generalization of our NER systems, we explore the prediction of masked tokens during the training as proposed by Cabrera-Diego et al. (2021). Firstly, this method converts randomly selected tokens, within a sentence, into BERT's special token *[MASK]*. Then, the NER system has to predict correctly the sentence's NEs, despite the missing information, as well as predicting the masked tokens. The prediction of masked tokens is done by introducing BERT's output into a linear layer, which has the same size of the pretrained vocabulary. During training, the loss produced by the prediction of masked tokens is added to the loss produced by the recognition of NEs; during testing, this layer is inactive.

Although Powalski and Stanislawek (2020) propose *UniCase*, an architecture for training a language model that learns the casing of a word separately to the tokenization, in this work, we use a simpler method that does not require to retrain a language model. Specifically, we use a marking and enrichment approach, where an uppercase word is tagged with two special BERT's tokens, defined by us as *[UP]* and *[up]*, and where we include additional case versions. For instance, the word "ROME" becomes "[UP] [ROM, ##E] [Rome] [r,##ome] [up]". It is important to indicate that the prediction of the NE type is done uniquely over the first token, which correspond to the special token [UP]. In other words, the output produced by BERT for the rest of the tokens is masked. The marking of the uppercase words is based on the ideas proposed by Cabrera-Diego et al. (2021).

4 Datasets

We use the data provided by the organizers for the 3rd Shared Task on SlavNER. However, for the

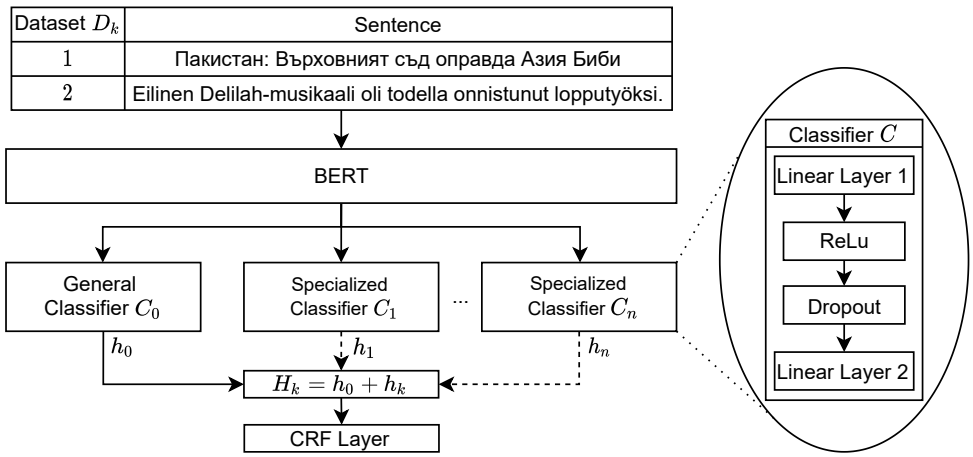


Figure 1: Our FEDA-based architecture for NER with BERT.

development of our internal models, we use the topics of *Nord Stream* and *Ryanair* as testing partition, while the rest as training and development. For the final models, all the data provided is split into training and development sets.

Besides the data provided by SlavNER's organizers, we use the follwing NER corpora:

SlavNER 2017 (Piskorski et al., 2017): Slavic Corpus annotated with 4 NE types: Location, Miscellaneous, Organization and Person.

Collection Named Entities 5 (CNE5) (Mozharova and Loukachevitch, 2016)[3]: Russian NER corpus manually annotated with five NE types: Geopolitical, Location, Media, Person and Organization.

Czech Named Entity Corpus 2.0 (CNEC) (Ševčíková et al., 2007): Czech corpus annotated with fine-grained NE. In this work, we have used 6 types of NE: Location, Organization, Media, Artifact, Person and Time.

FactRuEval[4]: Russian corpus annotated with three NE types: Location, Organization and Person.

Finnish NER (Luoma et al., 2020): Although Finnish is not a language to process in SlavNER, it has similar NE types to those used in the shared task: Date, Event, Location, Organization, Person, Product and Time. We use this dataset to enrich the NEs knowledge, specially on events and products.

National Corpus of Polish (NKJP)[5] (Przepiórkowski et al., 2012): Polish corpus tagged with five NE types: Person, Organization, Geopolitical, Location, Date and Time.

NER-UK[6]: Collection of 264 Ukrainian documents manually annotated with four types of NE: Location, Miscellaneous, Organization and Person.

Polish Corpus of Wrocław University of Technology (KPWr)[7] (Marcińczuk et al., 2016): Polish dataset annotated with nine super NE types, from these six were chosen: Event, Location, Organization, Person, Place and Product. Location and Place were merged as the former.

SSJ500k (Krek et al., 2019): Slovene corpus annotated with four types of NE: Location, Miscellaneous, Organization and Person.

Wikiann (Pan et al., 2017): It is a multilingual NER corpus based on Wikipedia articles; it was annotated automatically using three types of NEs: Location, Organization and Person. We use of the corpus partitions used by Rahimi et al. (2019).

We use for all the additional corpora their training, development and testing partitions; if these are not provided, we create them using a stratified approach to ensure a proportional number of NEs.

5 Experimental Setup

Regarding BERT, we use different pre-trained models: *CroSloEngual* (Ulčar and Robnik-Šikonja, 2020), *Polish BERT*[8], *RuBERT* (Kuratov and Arkhipov, 2019) and *Language-Agnostic BERT Sentence Embedding (LaBSE)* (Feng et al., 2020).

All the files coming from SlavNER are tokenized and, those used for training and development are annotated at token-level. For Bulgarian and Slovene, we tokenize the documents using Reldi-Tokenizer[9], while for the rest of languages, we use the neural parser proposed by Kanerva et al. (2018). Further-

[3] labinform.ru/pub/named_entities
[4] github.com/dialogue-evaluation/factRuEval-2016
[5] nkjp.pl
[6] github.com/lang-uk/ner-uk
[7] clarin-pl.eu/dspace/handle/11321/270
[8] huggingface.co/dkleczek/bert-base-polish-cased-v1
[9] github.com/clarinsi/reldi-tokeniser

more, we over-tokenize all the files, i.e. we separate all the punctuation from tokens within a sentence, to solve some cases where abbreviation periods or dashes were not considered as part of a NE. For example, in Slovene, Roman numerals are followed by a period, such as in *Benedikt XVI.* nevertheless, some NE annotations did not consider the period. Some rules and manual corrections were applied to the tokenization where we determined the fix was critical. For instance, in Polish, *W. Brytania* (Great Britain) was being split into two sentences by the tokenizer. We automatically annotated the files by searching the longest match in the tokenized format and the annotation file. In case of ambiguity, the annotation tool requested a manual intervention. For the final submission, we converted the token-level output to a document-level one.

All the NEs types are encoded using BIOES (Beginning, Inside, Outside/Other, End, Single). As well, to reduce the number of entities types, we normalize those where the theoretical meaning is the same, i.e. PERS into PER or EVENT into EVT.

For the models where masked tokens have to be predicted, we only affect sentences in the training partitions that are longer than 3 actual tokens, i.e. not BPE tokens. At each epoch, we select randomly 25% of each sentence's tokens and substitute them with *[MASK]*. If a token after being processed by BERT's tokenizer produces more than one BPE token, we mask only one of them.[10] Regarding the models that are trained with marked uppercase tokens, at each training epoch, we randomly convert 5% of all the sentences into uppercase. This is done to provide some examples of uppercase sentences to datasets that do not present this phenomenon.

In Table 2, we present the final models created for recognizing NEs. As well, we detail which are the datasets used for training them and which are the additional features that they make use. The combinations of datasets and features used for the final models were selected according to their performance on internal models. To enrich the knowledge in Bulgarian, we added the Macedonian Wikiann dataset, as both languages are considered as mutually intelligible. All the models were trained up to 20 epochs using an early stop approach. In Table 1, we present a summary of the hyperparameters used for training the NER systems.

[10]For Polish BERT, we mask all the tokens as this model was trained using whole word masking.

Hyperparameter	**Value**
Maximum Epochs	20
Early Stop Patience	2
Learning Rate	2×10^{-5}
Scheduler	Linear with warm-up
Warm-up Ratio	0.1
Optimizer	AdamW with bias correction
AdamW ϵ	1×10^{-8}
Random Seed	12
Dropout rate	0.5
Weight decay	0.01
Clipping gradient norm	1.0
BERT's Sequence Size	128
Linear Layer 1 Size	512
Training Mini-Batch:	
Latin 1 & 2	5
Ru	8
Pl	28
Others	16

Table 1: Hyperparameters used for training the models.

6 Results

In Table 3, we present the performance of our systems in terms of strict micro F-score. We can observe, that the marking of uppercase words worked better, in general, for the *Covid-19* topic, specially on the Cyrillic-2 model. As well, single language models worked better on the *Covid-19* topic, while the model Latin-1 worked better on the *U.S. Elections* topic. In most languages, the hardest NEs to predict were related to products and events due to their weak regularity or because they never appeared on the training datasets.

From a manual inspection, we have observed that multiple events were considered as products, such as *Miss USA*, *Pizzagate* and *Covid-19*. Some products were marked as organizations such as *Zoom*, *COVAX*, *Apple TV+*, although fewer organizations were tagged as products, such as *Pfizer/Moderna* and *BBC*. Nonetheless, many of these NEs could be both types depending on the context in which happen. In certain documents, organizations were marked as locations and viceversa, such as *Ostravské Fakultní Nemocnice* (Ostrava University Hospital) and *Szpitala Wojskowego w Szczecinie* (Military Hospital in Szczecin).

We have found interesting examples regarding products despite their irregularity. For example, the Cyrillic and Latin models managed to detect partially the 2020 book "*Nelojalen: resnična zgodba nekdanjega osebnega odvetnika predsednika Donalda Trumpa*" (Disloyal: A Memoir: The True Story of the Former Personal Attorney to President Donald J. Trump). Specifically, the entity was

Model		Features	BERT Model	*C*	Training datasets
Script-based	Cyrillic-1 Cyrillic-2	None Uppercase	LaBSE	8	SlavNER-17 (Ru, Uk); SlavNER-21 (Bg, Ru, Uk); Wikiann (Bg, Mk, Ru, Uk); FactRuEval; CNE5; NER-UK; Finnish NER
	Latin-1 Latin-2	None Uppercase	LaBSE	8	SlavNER-17 (Cs, Pl, Sl); SlavNER-21 (Cs, Pl, Sl); Wikiann (Cs, Pl, Sl); SSJ500k; KPWr; CNEC; Finnish NER
Single language	Bg	Uppercase	LaBSE	5	SlavNER-21 (Bg); Wikiann (Bg, Mk); Finnish NER
	Cs	Uppercase	LaBSE	5	SlavNER-21 (Cs); Wikiann (Cs); CNEC; Finnish NER
	Pl	Mask.+Upper.	Polish BERT	5	SlavNER-21 (Pl); Wikiann (Pl); KPWr; NKJP
	Ru	Mask.+Upper.	RuBERT	5	SlavNER-21 (Ru); Wikiann (Ru); FactRuEval; CNE5
	Sl	Mask.+Upper.	CroSloEngual	4	SlavNER-21 (Sl); Wikiann (Sl); SSJ500k
	Uk	Uppercase	LaBSE	4	SlavNER-21 (Uk); Wikiann (Uk); NER-UK

Table 2: Datasets used for training each of the model explored in this work. The number of classifiers (C) consider both the general and specialized ones used in the architecture.

	Covid-19							U.S. Elections							
Model	**Bg**	**Cs**	**Pl**	**Ru**	**Sl**	**Uk**	**All**	**Bg**	**Cs**	**Pl**	**Ru**	**Sl**	**Uk**	**All**	**Global**
Cyrillic-1	0.716	0.714	0.760	0.657	0.732	0.722	0.715	0.843	0.837	0.841	0.741	0.837	0.787	0.793	0.764
Cyrillic-2	0.720	0.730	0.783	0.642	0.744	**0.727**	0.721	**0.865**	0.857	0.849	0.746	0.858	**0.813**	0.807	0.775
Latin-1	0.730	0.765	0.791	0.662	0.752	0.706	0.733	0.850	**0.890**	**0.908**	**0.762**	**0.898**	0.789	**0.824**	**0.790**
Latin-2	**0.733**	0.763	0.792	**0.666**	0.758	0.688	**0.734**	0.854	**0.890**	0.891	0.759	0.884	0.782	0.819	0.787
Single lang.	0.725	**0.766**	**0.793**	0.611	**0.775**	0.701	0.729	0.813	0.889	0.887	0.742	0.891	0.781	0.807	0.778

Table 3: Strict micro F-scores obtained by each model for every language and topic. The *Global* column is the strict micro F-score regarding all the test data.

split into two "*Nelojalen: resnična zgodba nekdanjega osebnega odvetnika predsednika*" as a product and *Donalda Trumpa* (Donald Trump) as a person. But there were some exact matches, such as the book "*Cyberwar: How Russian Hackers and Trolls Helped Elect a President*" or the document "*Preveč in nikoli dovolj: kako je moja družina ustvarila najnevarnejšega moža na svetu*" (Treaty on Measures for the Further Reduction and Limitation of Strategic Offensive Arms). Furthermore, some scientific articles were tagged as products, such as "*A Study to Evaluate Efficacy, Safety, and Immunogenicity of mRNA-1273 Vaccine in Adults Aged 18 Years and Older to Prevent COVID-19*", although they did not appear in the gold standard.

Some models considered *BioNTech* as an organization and *Instagram* as a product despite these NEs were never seen during the training. As well, some medication-related products were correctly found such as *AZD1222*, канакинумаб (Canakinumab), *Remdesivir* or *Zithromax*, even if they did not exist on the training corpora.

We observed, specially in Cyrillic-scripted languages, that some named entities were incorrect because they were predicted without punctuation marks. For example: *Moderna Inc* vs *Moderna Inc.*, гам-ковид-вак vs «гам-ковид-вак» and спутником vs "спутником". In Latin-scripted languages, we observed the opposite although less frequently. For instance, *Roberta F. Kennedyho Jr.* vs *Roberta F. Kennedyho Jr*. In some documents the punctuation mark is included in certain NEs but not in others, such as in *Korea Płn.* vs *Korea Płn* but *Korei Płn.*.

7 Conclusions and Future Work

This work presented the participation of Laboratory *L3i* in the 3rd Shared Task on SlavNER. Specifically, we proposed a collection of BERT-based NER systems that were trained using multiple datasets through FEDA.

The results showed us that our NER systems worked better on the *U.S. Elections* topic (strict micro F-score between 0.762 and 0.908) than on the *Covid-19* topic (0.666 - 0.775). Overall, a competitive strength of our NER systems is that they managed to predict named entities occurring with weak regularity or that were never seen before.

In the future, we will apply the proposed architecture on other languages and datasets.

Acknowledgments

This work has been supported by the European Union's Horizon 2020 research and innovation program under grants 770299 (NewsEye) and 825153 (Embeddia).

References

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.

Luis Adrián Cabrera-Diego, Jose G. Moreno, and Antoine Doucet. 2021. Simple ways to improve NER in every language using markup. In *Proceedings of the 2nd International Workshop on Cross-lingual Event-centric Open Analytics*, Online. CEUR-WS.

Hal Daumé III. 2007. Frustratingly Easy Domain Adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic. Association for Computational Linguistics.

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.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. *Language-agnostic BERT Sentence Embedding*. ArXiv cs.CL eprint: 2007.01852.

Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. Turku Neural Parser Pipeline: An End-to-End System for the CoNLL 2018 Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics.

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2016. Frustratingly Easy Neural Domain Adaptation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 387–396, Osaka, Japan. The COLING 2016 Organizing Committee.

Simon Krek, Kaja Dobrovoljc, Tomaž Erjavec, Sara Može, Nina Ledinek, Nanika Holz, Katja Zupan, Polona Gantar, Taja Kuzman, Jaka Čibej, Špela Arhar Holdt, Teja Kavčič, Iza Škrjanec, Dafne Marko, Lucija Jezeršek, and Anja Zajc. 2019. Training corpus ssj500k 2.2. Slovenian language resource repository CLARIN.SI.

Yuri Kuratov and Mikhail Arkhipov. 2019. *Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language*. ArXiv cs.CL eprint: 1905.07213.

Bill Yuchen Lin, Dong-Ho Lee, Ming Shen, Ryan Moreno, Xiao Huang, Prashant Shiralkar, and Xiang Ren. 2020a. TriggerNER: Learning with Entity Triggers as Explanations for Named Entity Recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8503–8511, Online. Association for Computational Linguistics.

Hongyu Lin, Yaojie Lu, Jialong Tang, Xianpei Han, Le Sun, Zhicheng Wei, and Nicholas Jing Yuan. 2020b. A Rigorous Study on Named Entity Recognition: Can Fine-tuning Pretrained Model Lead to the Promised Land? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7291–7300, Online. Association for Computational Linguistics.

Jouni Luoma, Miika Oinonen, Maria Pyykönen, Veronika Laippala, and Sampo Pyysalo. 2020. A Broad-coverage Corpus for Finnish Named Entity Recognition. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4615–4624, Marseille, France. European Language Resources Association.

Michał Marcińczuk, Marcin Oleksy, Marek Maziarz, Jan Wieczorek, Dominika Fikus, Agnieszka Turek, Michał Wolski, Tomasz Bernaś, Jan Kocoń, and Paweł Kedzia. 2016. *Polish Corpus of Wrocław University of Technology 1.2*. CLARIN-PL digital repository.

Jose G. Moreno, Elvys Linhares Pontes, Mickaël Coustaty, and Antoine Doucet. 2019. TLR at BSNLP2019: A multilingual named entity recognition system. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 83–88, Florence, Italy. Association for Computational Linguistics.

Valerie Mozharova and Natalia Loukachevitch. 2016. Two-stage approach in Russian named entity recognition. In *2016 Conference on Intelligence, Social Media and Web (ISMW FRUCT)*, pages 1–6, St. Petersburg, Russia.

Debora Nozza, Pikakshi Manchanda, Elisabetta Fersini, Matteo Palmonari, and Enza Messina. 2021. LearningToAdapt with word embeddings: Domain adaptation of Named Entity Recognition systems. *Information Processing & Management*, 58(3):102537.

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, Vancouver, Canada. Association for Computational Linguistics.

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.

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, Valencia, Spain. Association for Computational Linguistics.

Rafal Powalski and Tomasz Stanislawek. 2020. *UniCase – Rethinking Casing in Language Models*. ArXiv cs.CL eprint: 2010.11936.

Adam Przepiórkowski, Mirosław Bańko, Rafał L. Górski, and Barbara Lewandowska-Tomaszczyk. 2012. *Narodowy korpus jezyka polskiego*. Naukowe PWN, Warsaw, Poland.

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.

Lichao Sun, Kazuma Hashimoto, Wenpeng Yin, Akari Asai, Jia Li, Philip Yu, and Caiming Xiong. 2020. *Adv-BERT: BERT is not robust on misspellings! Generating nature adversarial samples on BERT*. ArXiv cs.CL eprint: 2003.04985.

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.

Tatiana Tsygankova, Stephen Mayhew, and Dan Roth. 2019. BSNLP2019 Shared Task Submission: Multisource Neural NER Transfer. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 75–82, Florence, Italy. Association for Computational Linguistics.

Matej Ulčar and Marko Robnik-Šikonja. 2020. FinEst BERT and CroSloEngual BERT. In *Proceeding of the 23rd International Conference on Text, Speech, and Dialogue (TSD 2020)*, pages 104–111, Brno, Czech Republic. Springer International Publishing.

Magda Ševčíková, Zdeněk Žabokrtský, and Oldřich Krůza. 2007. Named Entities in Czech: Annotating Data and Developing NE Tagger. In *Text, Speech and Dialogue*, pages 188–195, Pilsen, Czech Republic. Springer Berlin Heidelberg.

B Atténuer les erreurs de numérisation dans la reconnaissance d'entités nommées pour les documents historiques

Atténuer les erreurs de numérisation dans la reconnaissance d'entités nommées pour les documents historiques

Emanuela Boros — Ahmed Hamdi[1] — Elvys Linhares Pontes[1] — Luis Adrián Cabrera-Diego[1] — Jose G. Moreno[1,2] — Nicolas Sidere[1] — Antoine Doucet[1]

[1] *Laboratoire L3i, Université de La Rochelle, France*
[2] *IRIT, Université de Toulouse, France*

RÉSUMÉ. *Cet article aborde la reconnaissance d'entités nommées (NER) appliquée aux textes historiques obtenus à partir du traitement d'images numériques de journaux à l'aide de techniques de reconnaissance optique de caractères (OCR). Nous soutenons que le principal défi pour cette tâche est que le processus OCR produit des textes contenant entre autres des fautes d'orthographe et des erreurs de syntaxes. De plus, des variations sémantiques peuvent être présentes dans les documents anciens, ce qui a un impact sur les performances de la reconnaissance d'entités nommées. Nous menons une évaluation comparative à l'état de l'art de deux ensembles de données historiques en allemand et en français, et nous proposons un modèle basé sur une pile hiérarchique de couches Transformer pour aborder la reconnaissance d'entités nommées dans des données historiques. Nos résultats montrent que le modèle proposé améliore clairement les résultats sur les deux ensembles de données.*

ABSTRACT. *This paper tackles the task of NER applied to historical texts obtained from processing digital images of newspapers using OCR techniques. The main challenge for this task is that the OCR process leads to misspellings and linguistic errors in the output text, which can impact the performance of the NER. We conduct a comparative evaluation on two historical datasets in German and French against previous state-of-the-art models, and we propose a model based on a hierarchical stack of Transformers to approach the NER task for historical data. Our findings show that the proposed model clearly improves the results on both historical datasets.*

MOTS-CLÉS : *Extraction d'information, reconnaissance d'entités nommées, données multilingues, données historiques.*

KEYWORDS: *Information extraction, Named entity recognition, Multilingual data, Historical data*

1. Introduction

Avec la numérisation à grande échelle de contenus patrimoniaux, le besoin de rendre efficacement accessible les documents historiques à l'aide de technologies appropriées a très fortement augmenté. Dans le même temps, il existe un intérêt croissant pour l'extraction d'informations pertinentes à partir de sources historiques. Dans cet article, nous abordons la tâche de la reconnaissance d'entités nommées (NER). qui vise à identifier des entités du monde réel, telles que les noms de personnes, d'organisations et de lieux à partir des textes bruts.

Alors que la plupart des travaux de recherche se concentrent sur les ensembles de données contemporains, les performances des systèmes NER ont augmenté à un rythme rapide, grâce à la capacité de représentation des réseaux de neurones. Plus récemment, les modèles NER basés sur des représentations contextuelles de mots et de chaînes de caractères fournis par Flair (Akbik *et al.*, 2018) ou BERT (Devlin *et al.*, 2019) ont permis des améliorations impressionnantes. Les architectures (Vaswani *et al.*, 2017) basées sur Transformer pour NER sont devenues populaires depuis la sortie du modèle BERT.

Pour extraire des entités de documents historiques, les outils NER sont confrontés à des défis supplémentaires. La majorité de ces documents est numérisée et traitée par un outil de reconnaissance optique de caractères (OCR) pour transcrire le texte. Cependant, la sortie de l'OCR peut potentiellement contenir des erreurs. Cela est principalement dû à la qualité de l'outil ou encore à la dégradation des documents numérisés en particulier pour les documents historiques. Cela conduit à des erreurs dans le texte transcrit, notamment des emplacements ou des noms de personnes mal orthographiés, ce qui est problématique puisque ce type d'entité nommée fait fréquemment partie des requêtes soumises aux collections patrimoniales. Pour relever ces défis nous proposons un modèle NER robuste basé sur une pile de *Transformers* qui comprend des encodeurs BERT affinés. Nous étudions l'impact d'un tel modèle, et nous concluons que ce type de modèle est adapté à l'extraction d'entités à partir de documents historiques. Le travail présenté ici est décrit plus en détail dans (Boroş *et al.*, 2020).

2. Ensembles de données

Des expériences ont été menées sur deux jeux de données issues de presse ancienne numérisée HIPE et NEWSEYE. Chaque ensemble propose deux corpus en français et en allemand. L'ensemble de données HIPE a été créé par le défi HIPE du laboratoire d'évaluation CLEF 2020 (Ehrmann *et al.*, 2020a). Il est composé d'articles de plusieurs journaux historiques suisses, luxembourgeois et américains publiés de 1790 à 2010 (Ehrmann *et al.*, 2020b).

Nous utilisons également l'ensemble de données NEWSEYE, composé de journaux historiques en français (1814-1944) et en allemand (1845-1945). Les documents ont été collectés auprès des bibliothèques nationales de France et d'Autriche (ONB), respectivement. HIPE et NEWSEYE utilisent des guides d'annotation similaires et com-

patibles entre eux. A l'exception de l'entité *TIME* qui est utilisé uniquement dans HIPE, toutes les autres classes sont identiques dans les deux jeux de données.

3. Modèle

D'abord, nous utilisons un modèle BERT pré-entraîné, et nous ajoutons ensuite n blocs *Transformer* par-dessus, finalisés avec une couche de prédiction CRF. Nous appelons ce modèle BERT $+n\times$Transf où n est un hyper-paramètre faisant référence au nombre de couches de *Transformer*.

Néanmoins, malgré l'impact majeur de BERT, les chercheurs s'interrogent sur la capacité de ce modèle à traiter des contenus bruités (Sun *et al.*, 2020) à moins que des techniques complémentaires ne soient utilisées (Muller *et al.*, 2019 ; Pruthi *et al.*, 2019). En plus de BERT, nous ajoutons ainsi une pile de blocs *Transformer* (Vaswani *et al.*, 2017) (encodeurs). Nous supposons que les couches *Transformer* complémentaires permettent d'atténuer la sensibilité du lemmatiseur intégré de BERT aux erreurs OCR tels que les mots hors vocabulaire (OOV) ou les fautes d'orthographe, et contribuer à l'apprentissage et à la reconnaissance du contexte des entités.

4. Éxpériences

Nous avons choisi comme base le modèle proposé par (Ma et Hovy, 2016), un modèle end-to-end combinant un encodage de caractères BiLSTM et CNN, afin de profiter des fonctionnalités de mots et de caractères[1]. L'analyse au niveau caractère est connue comme permettant de capturer des informations morphologiques et de forme (Kanaris *et al.*, 2007 ; Santos et Zadrozny, 2014 ; dos Santos et Guimarães, 2015).

L'évaluation de la tâche NER se fait avec le niveau entité comme unité de référence (Makhoul *et al.*, 1999). Nous calculons la précision (P), le rappel (R) et la mesure F1 (F1) au niveau micro, c'est-à-dire que les types d'erreur sont considérés sur tous les documents. Deux scénarios d'évaluation ont été considérés : *micro-strict*, qui recherche une correspondance exacte des entités, et *micro-fuzzy*, où une prédiction est correcte lorsqu'il y a au moins un chevauchement de tokens (Ehrmann *et al.*, 2020a). En outre, la significativité statistique est mesurée par un test t bilatéral, avec une valeur p estimée entre 0,01 et 0,05 (* dénote une amélioration significative par rapport au modèle d'avant à $p \leq 0{,}05$, ** dénote $p \leq 0{,}01$).

À partir des résultats de la table 1, nous pouvons voir la preuve que les modèles basés sur BERT avec $n\times$ Transf atteignent, pour les ensembles de données et les langues, des textit micro-fuzzy et textit micro-strict valeurs de performance que le modèle BERT autonome et les modèles de base. Tous les modèles ont une signification

1. Une description détaillée du modèle et des hyperparamètres peut être trouvée dans (Ma et Hovy, 2016).

	HIPE						NEWSEYE					
	DE			FR			DE			FR		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BiLSTM-CNN												
fuzzy	83.3	70.1	76.1	89.9	83.9	86.8	81.2	42.4	55.7	82.2	77.2	79.6
strict	69.4	58.4	63.4	77.7	72.5	75.0	54.8	28.6	37.6	65.5	61.4	63.4
BERT												
fuzzy	83.4	88.3	85.8**	89.5	91.9	90.7*	60.1	67.0	63.4**	86.1	81.8	83.9**
strict	74.1	78.5	76.2**	81.1	83.3	82.1*	46.8	52.2	49.4**	70.1	66.6	68.3**
BERT+1×Transf												
fuzzy	85.8	87.3	86.5**	91.3	92.9	**92.1****	82.3	66.4	**73.5****	88.7	82.1	**85.3****
strict	77.2	78.6	77.9**	83.5	84.9	**84.2****	62.7	50.6	56.0**	74.4	68.9	**71.5****
BERT+2×Transf												
fuzzy	87.0	87.2	**87.1****	91.5	92.4	91.9**	83.3	64.4	72.6**	89.7	80.1	84.7 **
strict	78.6	78.7	**78.7****	83.4	84.2	83.8**	64.9	50.2	**56.6****	75.0	67.0	70.8**

Tableau 1 : Résultats sur les ensembles de données HIPE et NEWSEYE en français et en allemand.

statistique $< 0,01$, ainsi, l'ajout de $n\times$ Transf peut améliorer la généralisabilité du modèle pour le NER sur les documents historiques.

De plus, ils parviennent généralement à maintenir un équilibre entre rappel et précision, alors que les modèles de référence varient selon la langue. On remarque également que, si en général les deux modèles obtiennent un équilibre entre rappel et précision, il existe un déséquilibre important dans le cas du jeu de données allemand NEWSEYE. BERT $+n\times$ Transf réduit la différence à 20 points, là où les méthodes de référence souffrent d'une différence de 40%.

5. Conclusions et perspectives

Nous avons présenté une architecture d'apprentissage profond pour le NER basé sur un encodeur BERT affiné et plusieurs blocs *Transformer*. Les résultats sur les deux jeux de données historiques en français et en allemand ont montré la capacité de l'approche proposée à traiter des corpus de textes numérisés bruités dans des langues distinctes. Si les améliorations apportées par le modèle NER proposé sont claires, notre analyse des résultats a mis en évidence plusieurs facteurs susceptibles d'influencer les résultats. Une analyse plus approfondie reste à mener. Nous comptons ainsi étudier les variations détaillées de notre architecture de manière plus approfondie.

Remerciements

Ce travail a été soutenu par le programme de recherche et d'innovation Horizon 2020 de l'Union européenne au titre des subventions 770299 (NewsEye) et 825153 (Embeddia).

6. Bibliographie

Akbik A., Blythe D., Vollgraf R., « Contextual string embeddings for sequence labeling », *Proceedings of the 27th International Conference on Computational Linguistics*, p. 1638-1649, 2018.

Boroş E., Hamdi A., Pontes E. L., Cabrera-Diego L.-A., Moreno J. G., Sidere N., Doucet A., « Alleviating Digitization Errors in Named Entity Recognition for Historical Documents », *Proceedings of the 24th Conference on Computational Natural Language Learning*, p. 431-441, 2020.

Devlin J., Chang M.-W., Lee K., Toutanova K., « BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding », *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)*, Association for Computational Linguistics, Minneapolis, Minnesota, p. 4171-4186, June, 2019.

dos Santos C., Guimarães V., « Boosting Named Entity Recognition with Neural Character Embeddings », *Proceedings of the Fifth Named Entity Workshop*, Association for Computational Linguistics, Beijing, China, p. 25-33, July, 2015.

Ehrmann M., Romanello M., Bircher S., Clematide S., « Introducing the CLEF 2020 HIPE Shared Task : Named Entity Recognition and Linking on Historical Newspapers », *European Conference on Information Retrieval*, Springer, p. 524-532, 2020a.

Ehrmann M., Romanello M., Clematide S., Ströbel P. B., Barman R., « Language Resources for Historical Newspapers : the Impresso Collection », *Proceedings of The 12th Language Resources and Evaluation Conference*, p. 958-968, 2020b.

Kanaris I., Kanaris K., Houvardas I., Stamatatos E., « Words versus character n-grams for anti-spam filtering », *International Journal on Artificial Intelligence Tools*, vol. 16, nº 06, p. 1047-1067, 2007.

Ma X., Hovy E., « End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF », *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, Association for Computational Linguistics, Berlin, Germany, p. 1064-1074, August, 2016.

Makhoul J., Kubala F., Schwartz R., Weischedel R. *et al.*, « Performance measures for information extraction », *Proceedings of DARPA broadcast news workshop*, Herndon, VA, p. 249-252, 1999.

Muller B., Sagot B., Seddah D., « Enhancing BERT for Lexical Normalization », *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 297-306, 2019.

Pruthi D., Dhingra B., Lipton Z. C., « Combating Adversarial Misspellings with Robust Word Recognition », *57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, Florence, Italy, p. 5582-5591, 2019.

Santos C. d., Zadrozny B., « Learning character-level representations for part-of-speech tagging », *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, p. 1818-1826, 2014.

Sun L., Hashimoto K., Yin W., Asai A., Li J., Yu P., Xiong C., « Adv-BERT : BERT is not robust on misspellings ! Generating nature adversarial samples on BERT », *arXiv preprint arXiv :2003.04985*, 2020.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I., « Attention is all you need », *Advances in neural information processing systems*, p. 5998-6008, 2017.

C Intérêt des modèles de caractères pour la détection d'événements

Intérêt des modèles de caractères pour la détection d'événements

Emanuela Boros[1] Romaric Besançon[2] Olivier Ferret[2] Brigitte Grau[3]
(1) La Rochelle Université, L3i, F-17042 La Rochelle
(2) Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
(3) Université Paris-Saclay, CNRS, LIMSI, ENSIIE, F-91405, Orsay, France
emanuela.boros@univ-lr.fr, romaric.besancon,olivier.ferret@cea.fr,
brigitte.grau@limsi.fr

RÉSUMÉ

Cet article aborde la tâche de détection d'événements, visant à identifier et catégoriser les mentions d'événements dans les textes. Une des difficultés de cette tâche est le problème des mentions d'événements correspondant à des mots mal orthographiés, très spécifiques ou hors vocabulaire. Pour analyser l'impact de leur prise en compte par le biais de modèles de caractères, nous proposons d'intégrer des plongements de caractères, qui peuvent capturer des informations morphologiques et de forme sur les mots, à un modèle convolutif pour la détection d'événements. Plus précisément, nous évaluons deux stratégies pour réaliser une telle intégration et montrons qu'une approche de fusion tardive surpasse à la fois une approche de fusion précoce et des modèles intégrant des informations sur les caractères ou les sous-mots tels que ELMo ou BERT.

ABSTRACT

The interest of character-level models for event detection.

This paper tackles the task of event detection that aims at identifying and categorizing event mentions in texts. One of the difficulties of this task is the problem of event mentions corresponding to misspelled, custom, or out-of-vocabulary words. To analyze the impact of character-level features, we propose to integrate character embeddings, which can capture morphological and shape information about words, to a convolutional model for event detection. More precisely, we evaluate two strategies for performing such integration and show that a late fusion approach outperforms both an early fusion approach and models integrating character or subword information such as ELMo or BERT.

MOTS-CLÉS : Extraction d'information, événements, plongements lexicaux.

KEYWORDS: Information extraction, events, word embeddings.

1 Introduction

Dans cet article, nous nous concentrons plus particulièrement sur la détection d'événements, qui implique l'identification d'instances de types d'événements prédéfinis dans un texte. Ces instances, appelées mentions d'événements ou déclencheurs d'événements, prennent la forme de mots ou d'expressions polylexicales évoquant un type d'événements de façon plus ou moins spécifique. Les approches les plus efficaces pour réaliser cette tâche sont actuellement fondées sur des modèles neuronaux (Chen *et al.*, 2015; Nguyen & Grishman, 2015; Nguyen *et al.*, 2016a,b; Feng *et al.*,

	Tous les mots	Mentions d'événements
entraînement	14 021	931
test	3 553	219
mots inconnus dans le test	930 (26,2%)	66 (30,1%)
mots inconnus avec un mot similaire	825	54

TABLE 1 – Statistiques concernant le vocabulaire des parties entraînement et test du corpus ACE 2005. Mot inconnu : présent dans la partie entraînement mais pas dans la partie test

2016; Zhang *et al.*, 2019; Nguyen & Grishman, 2018) et ont permis en particulier de s'affranchir du problème du choix des traits linguistiques utilisés par les modèles d'apprentissage statistiques. Ces modèles reposent ainsi sur des plongements de mots qui les rendent en principe moins sensibles au problème des déclencheurs non rencontrés lors de l'entraînement puisque ces plongements intègrent une forme de similarité entre les mots.

Toutefois, cette capacité peut varier en fonction des raisons pour lesquelles un déclencheur n'a pas été vu lors de l'entraînement du modèle. Nous illustrons ces différents cas sur la partie anglaise du jeu de données ACE 2005, un corpus standard pour l'évaluation de la détection d'événements dont nous reprenons la subdivision classiquement faite pour cette tâche entre entraînement, validation et test (Ji *et al.*, 2008). Le déclencheur inédit peut ainsi être une variante morphologique d'un déclencheur déjà vu dans l'ensemble des données d'entraînement. Par exemple, *torturing* n'est pas présent dans les données d'entraînement ACE 2005 mais il s'agit d'une variante de *torture*, qui est considéré comme un déclencheur pour le même type d'événements, en l'occurrence *Life.Injury*. En outre, *torturing* est susceptible d'être présent au sein d'un modèle de langue général, auquel cas un modèle de détection d'événements neuronal reposant sur ledit modèle de langue est susceptible de détecter avec succès ce déclencheur.

La situation est différente lorsqu'un déclencheur est absent des données d'entraînement parce qu'il correspond à une version mal orthographiée d'un déclencheur de référence. En effet, dans un tel cas, le modèle de langue ne contient pas nécessairement la version altérée. Par exemple, *aquitted* fait partie du corpus de test ACE 2005 pour référer à un événement *Justice.Sentence* alors que seule *acquitted*, la forme correcte pour ce mot, est présente dans les données d'entraînement. Dans ce cas, il est peu probable que le mot inédit fasse partie du modèle de langue général et, par conséquent, il a peu de chances d'être détecté comme déclencheur d'un événement *Justice.Sentence*. Plus globalement, comme le montre le tableau 1, 30,1 % des déclencheurs du corpus de test ACE 2005 ne sont pas présents dans le corpus d'entraînement mais 88 % de ces déclencheurs absents sont proches (mesurés par un ratio de Levenshtein inférieur à 0,3) de mots du corpus d'entraînement. Le tableau 2 présente des exemples de telles paires de mots. On peut voir qu'en dehors des paires correspondant à des différences de casse (intifada/Intifada) ou relevant de la morphologie flexionnelle (opening/open), certaines paires correspondent à des cas plus complexes relevant de la morphologie dérivationnelle (creating/creation) ou même de relations sémantiques complexes (hacked/attacked) qui ne sont souvent pas capturées par les modèles de plongements de mots.

Différentes stratégies ont été proposées pour traiter le problème de la variabilité lexicale dans les modèles de langue neuronaux. Pour les plongements statiques de mots, fastText (Bojanowski *et al.*, 2017) s'appuie ainsi sur une représentation des mots fondée sur des n-grammes de caractères. Pour les modèles contextuels, ELMo (Peters *et al.*, 2018) exploite une représentation fondée sur les caractères

Type d'événements	Déclencheur inconnu/connu le plus proche
Start-Org	*creating/creation, opening/open, forging/forming, formed/form*
End-Org	*crumbled/crumbling, dismantling/dismantle, dissolved/dissolving*
Transport	*fleeing/flying, deployment/deployed, evacuating/evacuated*
Attack	*intifada/Intifada, smash/smashed, hacked/attacked, wiped/wipe*
End-Position	*retirement/retire, steps/step, previously/previous, formerly/former*

TABLE 2 – Exemples de déclencheurs événementiels de test proches de déclencheurs d'entraînement

construite grâce à un réseau de neurones convolutif (CNN) tandis que BERT (Devlin *et al.*, 2019) adopte une stratégie mixte fondée sur des sous-mots, appelés wordpieces (Luong & Manning, 2016; Kim *et al.*, 2016; Jozefowicz *et al.*, 2016), avec quelques limites sur sa capacité à gérer les entrées bruitées (Sun *et al.*, 2020).

Nos contributions dans cet article sont plus particulièrement axées sur l'intégration de modèles reposant sur le niveau des caractères dans les modèles de détection d'événements pour traiter la question des mots inconnus. Plus précisément, nous montrons qu'un modèle de détection d'événements exploitant une représentation fondée sur les caractères est complémentaire d'un modèle fondé sur les mots et que leur combinaison selon une approche de fusion tardive est plus performante qu'une stratégie de fusion précoce.

2 Modèles

Notre approche s'inscrit dans le droit fil de la plupart des modèles de détection supervisée d'événements en considérant cette tâche comme une forme de classification multiclasse de mots : étant donné une phrase et un ensemble de types d'événements possibles, l'objectif est de prédire pour chacun de ses mots s'il relève ou non d'un de ces types d'événements et le cas échéant, duquel. L'entrée du système est donc un mot cible dans le contexte d'une phrase et sa sortie, un type d'événements ou l'étiquette NONE pour les mots non déclencheurs. Pour étudier l'influence des traits fondés sur les caractères, nous nous appuyons sur le modèle CNN proposé par Nguyen & Grishman (2015). Ce modèle de base est utilisé dans les deux composantes de notre modèle global : le modèle fondé sur les mots, dit modèle CNN mot, et le modèle fondé sur les caractères, dit modèle CNN caractère. Ces deux composantes sont combinées en utilisant soit une approche de fusion précoce, soit une approche de fusion tardive, comme l'illustre la figure 1.

Dans le modèle CNN mot, le contexte d'un mot candidat en tant que mention événementielle est formé par les mots qui l'entourent dans la phrase. Pour tenir compte de la nécessité de gérer des entrées de même dimension, ce contexte prend la forme d'une fenêtre de taille fixe, centrée sur la mention candidate. De ce fait, les parties de phrases dépassant la limite de cette fenêtre sont tronquées tandis qu'un remplissage avec des valeurs nulles (*zero-padding*) est réalisé pour les phrases plus courtes. Au sein de cette fenêtre de contexte, chaque mot est représenté par un plongement de mot et une position relative par rapport à la mention candidate, elle aussi sous la forme d'un plongement. Les plongements de mots et de positions sont concaténés et passés au travers d'une couche de convolution. Plus précisément, un ensemble de filtres convolutifs de tailles différentes sont appliqués et une opération de *max pooling* est appliquée à l'échelle de la fenêtre pour obtenir une

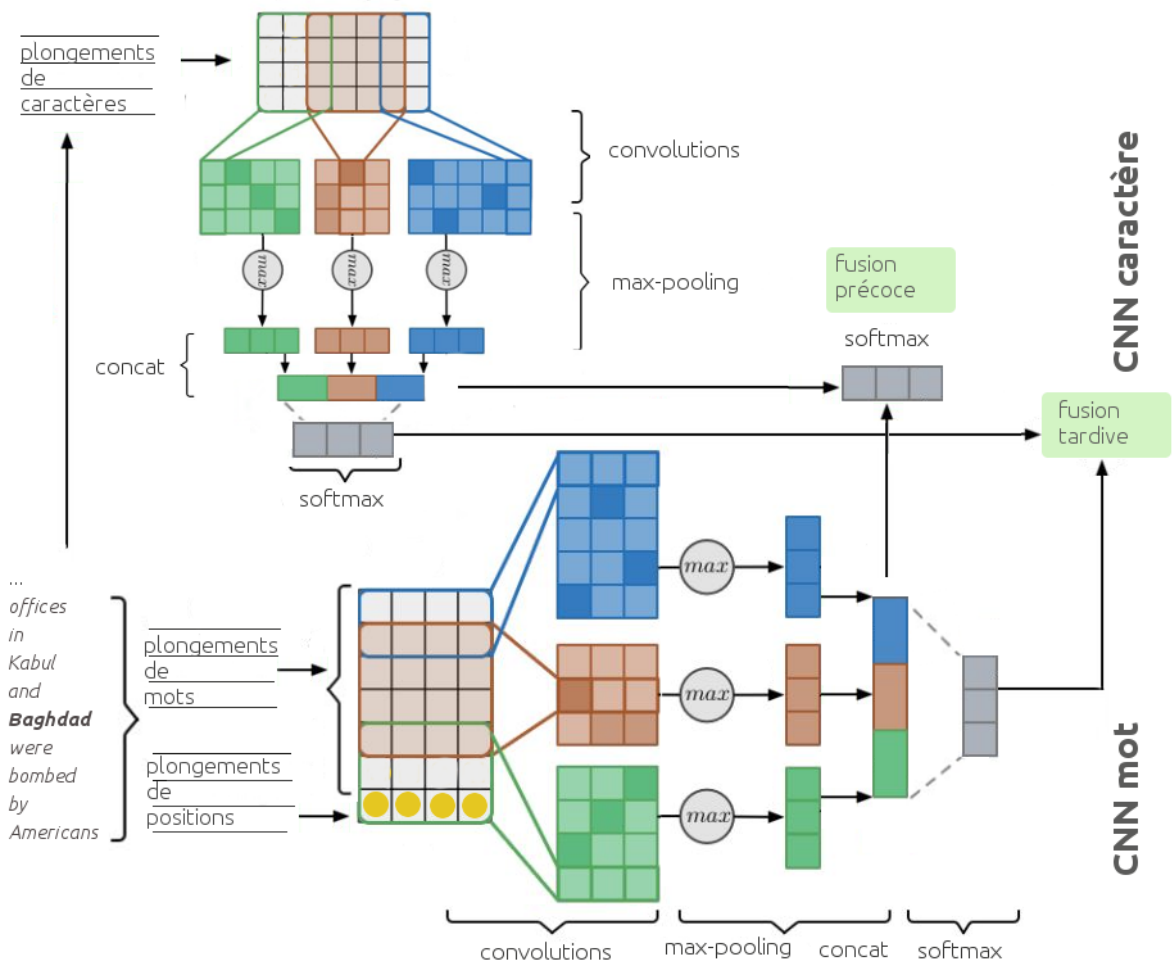


FIGURE 1 – Association d'un modèle fondé sur les mots et d'un modèle fondé sur les caractères

valeur par filtre. Le résultat de ces opérations se voit ensuite appliquer un *softmax* pour réaliser la classification en tant que telle. Le modèle CNN caractère est très proche du modèle CNN mot, avec deux différences principales : les mots sont remplacés par des caractères et il n'y a pas d'information de position associée à chaque caractère. Plus précisément, chaque mention candidate, identifiée sur la base des mots, se voit associer une fenêtre de contexte, comme dans le cas du CNN mot, mais cette fenêtre est dans ce cas déterminée sur la base d'un nombre fixe de caractères et les éléments de base de représentation sont constitués par des plongements de caractères. Les mêmes mécanismes de troncation et de remplissage permettant de considérer des phrases de taille variable et une fenêtre de contexte de taille fixe sont appliqués, mais ici à l'échelle du caractère.

Le premier type d'intégration de ces deux modèles est une fusion précoce, dans laquelle les deux représentations de la séquence d'entrée produites par les CNN de mots et de caractères sont concaténées avant la couche de classification. L'utilisation de ce type d'intégration permet un apprentissage conjoint des paramètres des deux modèles lors de la phase d'entraînement. L'intégration par fusion tardive repose quant à elle sur la combinaison par vote des décisions des deux modèles, qui sont entraînés séparément et apprennent donc des caractéristiques différentes des mentions candidates. La méthode de vote se définit comme suit : si une mention événementielle est détectée par un seul des deux modèles, nous conservons l'étiquette donnée par ce modèle ; sinon, si une mention est détectée par le CNN mot et le CNN caractère ensemble, nous conservons l'étiquette donnée par le CNN caractère. Cette stratégie est motivée par le fait que le modèle CNN mot possède une bonne couverture tandis que le modèle CNN caractère est davantage axé sur la précision.

3 Expérimentations, résultats et discussion

Cadre expérimental Nos expérimentations ont été réalisées sur le corpus ACE 2005. À des fins de comparabilité, nous utilisons le même découpage que les travaux antérieurs (Ji *et al.*, 2008; Liao & Grishman, 2010; Li *et al.*, 2013; Nguyen & Grishman, 2015; Nguyen *et al.*, 2016a), avec 529 documents (14 849 phrases) pour l'entraînement, 30 documents (863 phrases) pour le développement et 40 documents (672 phrases) pour le test. De même, nous considérons qu'une mention d'événement est correcte si son type d'événement, son sous-type et son empan correspondent à ceux d'une mention de référence. Nous utilisons les micro-mesures de précision, rappel et F1-mesure (F1) pour évaluer la performance globale.

Paramètres des modèles Pour le CNN mot, la taille de la fenêtre de contexte est de 31 mots. Les filtres de convolution ont pour leur part une dimension de 1, 2 et 3 mots et 300 filtres sont utilisés pour chaque dimension. Après chaque couche convolutive, initialisée selon un schéma orthogonal (Saxe *et al.*, 2014), une couche non linéaire *ReLU* est appliquée. Nous employons un abandon (*dropout*) de probabilité 0,5 après la couche initiale des plongements et de probabilité 0,3 après la concaténation du résultat des convolutions. La dimensionnalité des plongements de positions est de 50, à l'instar de (Nguyen & Grishman, 2015). Enfin, nous avons utilisé les plongements de mots préentraînés construits avec Word2vec sur le corpus Google News (Mikolov *et al.*, 2013).

Pour le CNN caractère, l'entrée est constituée de séquences de 1 024 caractères. Nous considérons tous les caractères sauf l'espace. La taille des filtres de convolution va de 2 à 10, avec 300 filtres par taille. La non-linéarité et l'initialisation de la couche convolutive sont les mêmes que pour le CNN mot. Les plongements de caractères comportent 300 dimensions et sont initialisés sur la base d'une distribution normale. Un abandon de 0,5 est réalisé après les plongements de caractères. Lors de l'entraînement conjoint dans le modèle de fusion précoce, les vecteurs de traits obtenus après les convolutions des deux modèles sont concaténés et comme pour le CNN mot, un abandon de 0,3 est appliqué avant la couche softmax.

Résultats et discussion Nous comparons notre modèle avec plusieurs modèles neuronaux proposés pour la même tâche n'utilisant pas de ressources externes : des modèles convolutifs (Nguyen & Grishman, 2015; Chen *et al.*, 2015; Nguyen *et al.*, 2016b; Nguyen & Grishman, 2018), des modèles récurrents (Nguyen *et al.*, 2016a; Zhao *et al.*, 2018), des modèles hybrides (Feng *et al.*, 2016), le modèle GAIL-ELMo (Zhang *et al.*, 2019) et un modèle fondé sur un mécanisme d'attention multilingue (Liu *et al.*, 2018). Nous ne considérons pas pour des raisons de comparabilité les modèles utilisant des ressources externes tels que (Bronstein *et al.*, 2015; Li *et al.*, 2019) ou (Yang *et al.*, 2019). Nous nous comparons également aux modèles plus récents fondés sur BERT tels que le modèle de (Wadden *et al.*, 2019) conjuguant BERT et un LSTM pour capturer un contexte intra et inter-phrastique et définir de façon plus dynamique les mentions candidates, le modèle BERT-QA (Du & Cardie, 2020), qui aborde la détection d'événements comme une tâche de question-réponse et le modèle DMBERT (Wang *et al.*, 2019), qui s'appuie sur l'apprentissage adverse pour mettre en œuvre une approche faiblement supervisée. Nous comparons également notre modèle avec 4 approches de base reposant sur BERT, en abordant la détection d'événements de manière similaire à la reconnaissance d'entités nommées dans (Devlin *et al.*, 2019) et avec les mêmes valeurs d'hyperparamètres.

La meilleure performance (F1 = 75,8 %) est obtenue en combinant les plongements de mots et de positions avec les plongements de caractères selon une stratégie de fusion tardive. Le tableau 3 montre également que l'ajout de plongements de caractères dans une stratégie de fusion tardive est plus performant que tous les modèles s'appuyant sur les mots, y compris les architectures complexes

Approches	**Précision**	**Rappel**	**F1**
Word CNN (Nguyen & Grishman, 2015)	71,8	66,4	69,0
Dynamic multi-pooling CNN (Chen *et al.*, 2015)	75,6	63,6	69,1
Joint RNN (Nguyen *et al.*, 2016a)	66,0	73,0	69,3
CNN with document context (Duan *et al.*, 2017)†	77,2	64,9	70,5
Non-Consecutive CNN (Nguyen *et al.*, 2016b)	na	na	71,3
Attention-based (Liu *et al.*, 2017)+	78,0	66,3	71,7
GAIL-ELMo (Zhang *et al.*, 2019)	74,8	69,4	72,0
Gated Cross-Lingual Attention (Liu *et al.*, 2018)	78,9	66,9	72,4
Graph CNN (Nguyen & Grishman, 2018)	77,9	68,8	73,1
Hybrid NN (Feng *et al.*, 2016)	84,6	64,9	73,4
DEEB-RNN3 (Zhao *et al.*, 2018)	72,3	75,8	74,0
BERT-base-uncased + LSTM (Wadden *et al.*, 2019)	na	na	68,9
BERT-base-uncased (Wadden *et al.*, 2019)	na	na	69,7
BERT-base-uncased (Du & Cardie, 2020)	67,2	73,2	70,0
BERT-QA (Du & Cardie, 2020)	71,1	73,7	72,4
DMBERT (Wang *et al.*, 2019)	77,6	71,8	74,6
DMBERT+Boot (Wang *et al.*, 2019)	77,9	72,5	75,1
BERT-base-uncased	71,7	68,5	70,0
BERT-base-cased	71,3	72,0	71,7
BERT-large-uncased	72,1	72,9	72,5
BERT-large-cased	69,3	**77,2**	73,1
CNN mot (équivalent à Word CNN)	71,4	65,9	68,5
CNN caractère	71,7	41,2	52,3
CNN mot + caractère - fusion précoce	**88,6**	61,9	72,9
CNN mot + caractère - fusion tardive	87,2	67,1	**75,8**

TABLE 3 – Évaluation de nos modèles et comparaison avec l'état de l'art pour la détection d'événements sur le test d'ACE 2005. †au-delà de la phrase, +avec les arguments de référence

s'appuyant sur les convolutions de graphe et les modèles exploitant BERT. Parmi ceux-ci, il est intéressant de noter que les modèles intégrant la casse (*cased*) sont plus performants que les modèles *uncased*, ce qui confirme l'importance de l'information portée par le niveau des caractères pour cette tâche, peut-être parce que la capitalisation est liée à la reconnaissance des entités nommées, qui sont généralement considérées comme importantes pour la détection des mentions d'événements. La similitude de nos résultats pour *BERT-base-uncased* avec ceux de (Du & Cardie, 2020) et (Wadden *et al.*, 2019) pour le même BERT accrédite par ailleurs la solidité de ce constat.

Cependant, nous pouvons constater que les plongements de caractères ne sont pas suffisants en eux-mêmes : en utilisant uniquement le CNN caractère, nous obtenons ainsi le plus petit rappel de toutes les approches considérées. Néanmoins, sa précision (71,7) est comparativement très élevée, ce qui confère une bonne fiabilité aux mentions qu'il détecte. Dans le cas de la fusion précoce, nous constatons que la précision est la plus élevée de tous les modèles comparés. Nous supposons que dans l'approche conjointe, l'influence des représentations fondées sur les caractères dépasse celle des plongements de mots et de positions et que la combinaison reproduit le déséquilibre entre la

Type d'événements	Nouvelles mentions trouvées	Mentions d'entraînement
End-Position	*steps*	*step*
Extradite	*extradited*	*extradition*
Attack	*wiped*	*wipe*
Start-Org	*creating*	*create*
Attack	*smash*	*smashed*
End-Position	*retirement*	*retire*

TABLE 4 – Nouvelles mentions trouvées grâce au modèle CNN mot+caractère (fusion tardive)

précision et le rappel observé pour le CNN caractère, le rappel étant le plus faible de tous les modèles à l'exception du CNN caractère. La fusion tardive permet un contrôle plus informé de la combinaison et, en donnant la priorité au CNN caractère pour déterminer le type des mentions identifiées par le CNN mot, la méthode tire profit de sa grande précision, permettant une augmentation de la précision de 71,7 à 87,2 tout en ayant un rappel élevé, passant de 65,9 pour le CNN mot à 67,1.

Finalement, nous avons mené une analyse plus qualitative en examinant les mentions d'événements nouvellement détectées par le modèle à fusion tardive comparativement au modèle à fusion précoce. Nous avons observé que parmi les 37 mentions concernées, certaines sont effectivement des variantes dérivationnelles ou flexionnelles de mots présents dans les données d'entraînement, comme illustré par le tableau 4. Ce constat semble confirmer que le modèle fondé sur les caractères peut capturer certaines informations sémantiques associées aux caractéristiques morphologiques des mots et parvenir ainsi à détecter de nouvelles mentions d'événements en relation avec des mentions d'entraînement. La présence dans le CNN caractère de filtres convolutifs d'une taille entre 2 et 10, c'est-à-dire couvrant une plage assez large de n-grammes de caractères, contribue très certainement à cette capacité.

4 Conclusion et perspectives

Dans cet article, nous avons étudié l'intégration de plongements de caractères dans un modèle neuronal de détection d'événements fondé un simple modèle CNN en testant des stratégies de fusion précoce ou tardive. Les meilleurs résultats sont obtenus en combinant les représentations fondées sur les mots avec celles fondées sur les caractères dans une stratégie de fusion tardive donnant la priorité au modèle de caractères pour décider du type d'événements. Cette méthode est plus performante que des approches plus complexes fondées sur les convolutions de graphe, les réseaux antagonistes ou les modèles BERT. Ces résultats montrent aussi qu'un modèle de caractères permet de surmonter certains problèmes concernant les mots nouveaux ou mal orthographiés dans les données de test.

Ce travail ouvre la voie à des études plus larges sur le problème de la robustesse des modèles de détection d'événements vis-à-vis des variations touchant les déclencheurs événementiels. De ce point de vue, il serait intéressant de tester si des modèles de langue de type Transformer s'appuyant sur les caractères (El Boukkouri *et al.*, 2020; Ma *et al.*, 2020), ou même s'affranchissant de la segmentation en mots (Clark *et al.*, 2021), pourraient s'avérer plus robustes qu'un modèle de type BERT.

Remerciements Ce travail a été partiellement soutenu par le programme européen Horizon 2020 au travers des projet NewsEyes (770299) et Embeddia (825153).

Références

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, **5**, 135–146.

BRONSTEIN O., DAGAN I., LI Q., JI H. & FRANK A. (2015). Seed-Based Event Trigger Labeling : How far can event descriptions get us ? In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 372–376.

CHEN Y., XU L., LIU K., ZENG D. & ZHAO J. (2015). Event extraction via dynamic multi-pooling convolutional neural networks. In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 167–176.

CLARK J. H., GARRETTE D., TURC I. & WIETING J. (2021). Canine : Pre-training an efficient tokenization-free encoder for language representation. *arXiv preprint arXiv :1602.02410*.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT 2019)*, p. 4171–4186, Minneapolis, Minnesota : Association for Computational Linguistics.

DU X. & CARDIE C. (2020). Event Extraction by Answering (Almost) Natural Questions. In *2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 671–683, Online.

DUAN S., HE R. & ZHAO W. (2017). Exploiting Document Level Information to Improve Event Detection via Recurrent Neural Networks. In *Eighth International Joint Conference on Natural Language Processing (IJCNLP 2017)*, p. 352–361 : Asian Federation of Natural Language Processing.

EL BOUKKOURI H., FERRET O., LAVERGNE T., NOJI H., ZWEIGENBAUM P. & TSUJII J. (2020). CharacterBERT : Reconciling ELMo and BERT for word-level open-vocabulary representations from characters. In *28^{th} International Conference on Computational Linguistics (COLING 2020)*, p. 6903–6915, Barcelona, Spain (Online : International Committee on Computational Linguistics.

FENG X., HUANG L., TANG D., JI H., QIN B. & LIU T. (2016). A language-independent neural network for event detection. In *54th Annual Meeting of the Association for Computational Linguistics*, p. 66–71.

JI H., GRISHMAN R. *et al.* (2008). Refining Event Extraction through Cross-Document Inference. In *46th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, p. 254–262.

JOZEFOWICZ R., VINYALS O., SCHUSTER M., SHAZEER N. & WU Y. (2016). Exploring the limits of language modeling. *arXiv preprint arXiv :1602.02410*.

KIM Y., JERNITE Y., SONTAG D. & RUSH A. M. (2016). Character-Aware Neural Language Models. In *Thirtieth AAAI Conference on Artificial Intelligence*, p. 2741–2749.

LI Q., JI H. & HUANG L. (2013). Joint Event Extraction via Structured Prediction with Global Features. In *51st Annual Meeting of the Association for Computational Linguistics*, p. 73–82.

LI W., CHENG D., HE L., WANG Y. & JIN X. (2019). Joint event extraction based on hierarchical event schemas from FrameNet. *IEEE Access*, **7**, 25001–25015.

LIAO S. & GRISHMAN R. (2010). Using document level cross-event inference to improve event extraction. In *48th Annual Meeting of the Association for Computational Linguistics*, p. 789–797 : Association for Computational Linguistics.

LIU J., CHEN Y., LIU K. & ZHAO J. (2018). Event Detection via Gated Multilingual Attention Mechanism. In *Thirty-second AAAI Conference on Artificial Intelligence (AAAI-18)*.

LIU S., CHEN Y., LIU K. & ZHAO J. (2017). Exploiting Argument Information to Improve Event Detection via Supervised Attention Mechanisms. In *55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, p. 1789–1798, Vancouver, Canada.

LUONG M.-T. & MANNING C. D. (2016). Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. In *54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, p. 1054–1063, Berlin, Germany.

MA W., CUI Y., SI C., LIU T., WANG S. & HU G. (2020). CharBERT : Character-aware pre-trained language model. In *28th International Conference on Computational Linguistics (COLING 2020)*, p. 39–50, Barcelona, Spain (Online) : International Committee on Computational Linguistics. DOI : 10.18653/v1/2020.coling-main.4.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. In *International Conference on Learning Representations (ICLR 2013), workshop track*.

NGUYEN T. H., CHO K. & GRISHMAN R. (2016a). Joint Event Extraction via Recurrent Neural Networks. In *2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 300–309.

NGUYEN T. H., FU L., CHO K. & GRISHMAN R. (2016b). A two-stage approach for extending event detection to new types via neural networks. *1st Workshop on Representation Learning for NLP*, p. 158.

NGUYEN T. H. & GRISHMAN R. (2015). Event Detection and Domain Adaptation with Convolutional Neural Networks. In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 365–371.

NGUYEN T. H. & GRISHMAN R. (2018). Graph Convolutional Networks With Argument-Aware Pooling for Event Detection. In *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*.

PETERS M., NEUMANN M., IYYER M., GARDNER M., CLARK C., LEE K. & ZETTLEMOYER L. (2018). Deep Contextualized Word Representations. In *2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT 2018)*, p. 2227–2237, New Orleans, Louisiana : Association for Computational Linguistics.

SAXE A. M., MCCLELLAND J. L. & GANGULI S. (2014). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *2nd International Conference on Learning Representations (ICLR 2014*.

SUN L., HASHIMOTO K., YIN W., ASAI A., LI J., YU P. & XIONG C. (2020). Adv-BERT : BERT is not robust on misspellings ! Generating nature adversarial samples on BERT. *arXiv preprint arXiv :2003.04985*.

WADDEN D., WENNBERG U., LUAN Y. & HAJISHIRZI H. (2019). Entity, relation, and event extraction with contextualized span representations. In *2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019)*, p. 5784–5789, Hong Kong, China.

WANG X., HAN X., LIU Z., SUN M. & LI P. (2019). Adversarial training for weakly supervised event detection. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT 2019)*, p. 998–1008.

YANG S., FENG D., QIAO L., KAN Z. & LI D. (2019). Exploring Pre-trained Language Models for Event Extraction and Generation. In *57th Annual Meeting of the Association for Computational Linguistics*, p. 5284–5294.

ZHANG T., JI H. & SIL A. (2019). Joint entity and event extraction with generative adversarial imitation learning. *Data Intelligence*, **1**(2), 99–120.

ZHAO Y., JIN X., WANG Y. & CHENG X. (2018). Document embedding enhanced event detection with hierarchical and supervised attention. In *56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, p. 414–419 : Association for Computational Linguistics.

D MELHISSA: A Multilingual Entity Linking Architecture for Historical Press Articles

International Journal on Digital Libraries (IJDL) manuscript No.
(will be inserted by the editor)

MELHISSA: A Multilingual Entity Linking Architecture for Historical Press Articles

Elvys Linhares Pontes · Luis Adrián Cabrera-Diego · Jose G. Moreno · Emanuela Boros · Ahmed Hamdi · Antoine Doucet · Nicolas Sidere · Mickaël Coustaty

Received: date / Accepted: date

Abstract Digital libraries have a key role in cultural heritage as they provide access to our culture and history by indexing books and historical documents (newspapers and letters). Digital libraries use natural language processing (NLP) tools to process these documents and enrich them with meta-information, such as named entities. Despite recent advances in these NLP models, most of them are built for specific languages and contemporary documents that are not optimized for handling historical material that may for instance contain language variations and optical character recognition (OCR) errors. In this work, we focused on the entity linking (EL) task that is fundamental to the indexation of documents in digital libraries. We developed a Multilingual Entity Linking architecture for HIstorical preSS Articles (MELHISSA) that is composed of multilingual analysis, OCR correction, and filter analysis to alleviate the impact of historical documents in the EL task. Experimentation has been done over two historical documents covering five European languages (English, Finnish, French, German, and Swedish). Results have shown that our system improved the global performance for all languages and datasets by achieving an F-score@1 of up to 0.655 and an F-score@5 of up to 0.752.

Keywords Entity linking · Historical data · Digital libraries · Deep learning · Heuristics

Elvys Linhares Pontes
Trading Central Labs, Sophia Antipolis, France
E-mail: elvyslpontes@gmail.com

Luis Adrián Cabrera-Diego
L3i, La Rochelle Université, La Rochelle, France
E-mail: luis.cabrera_diego@univ-lr.fr

Jose G. Moreno
University of Toulouse, IRIT, Toulouse, France
E-mail: jose.moreno@irit.fr

Emanuela Boros
L3i, La Rochelle Université, La Rochelle, France
E-mail: emanuela.boros@univ-lr.fr

Ahmed Hamdi
L3i, La Rochelle Université, La Rochelle, France
E-mail: ahmed.hamdi@univ-lr.fr

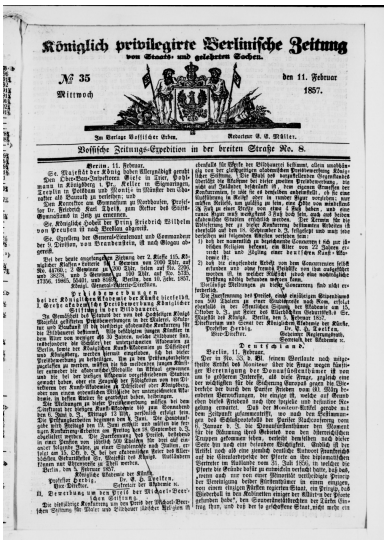
Antoine Doucet
L3i, La Rochelle Université, La Rochelle, France
E-mail: antoine.doucet@univ-lr.fr

Nicolas Sidere
L3i, La Rochelle Université, La Rochelle, France
E-mail: nicolas.sidere@univ-lr.fr

Mickaël Coustaty
L3i, La Rochelle Université, La Rochelle, France
E-mail: mickael.coustaty@univ-lr.fr

1 Introduction

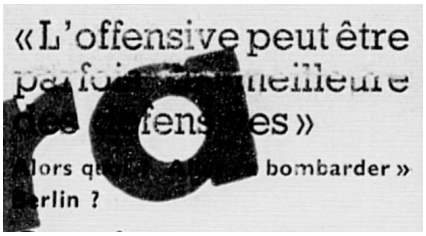
Historical documents are an essential resource in the understanding of our cultural heritage. The development of recent technologies, such as optical character recognition (OCR) systems, eases the digitization of physical documents and the extraction of textual content. Digitization provides two major advantages, in particular for digital humanities (DH) scholars: the exponential increase of target audiences, and the preservation of original documents from any damage when accessing them [44,9,53,27]. The recent interest in massive digitization raises multiple challenges to content providers including indexing, categorization, searching, to mention a few. Although these challenges also exist when dealing with contemporary text documents, digitized version augments each challenge because of inherent problems associated with the source quality (natural degradation of the documents) and to the digitiza-



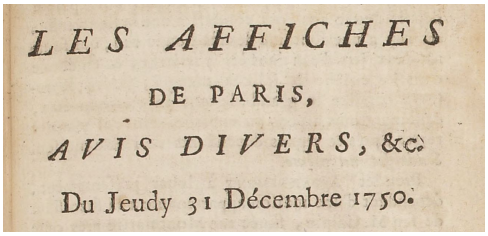
(a) 1857 German newspaper [3].

(b) 1890 American newspaper [4].

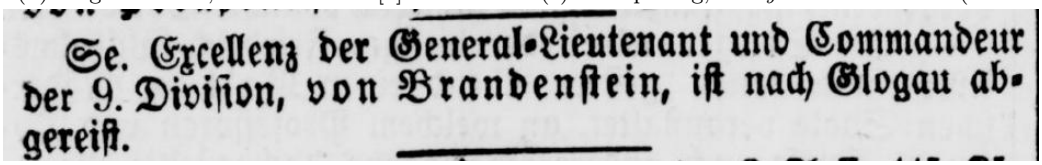
(c) 1936 French newspaper [5].



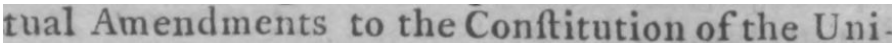
(d) Illegible words, such as *Berlin* [5].



(e) Old spelling, *Jeudy* instead of *Jeudi* (Thrusday) [6].



(f) Franktur font, which might be hard to recognize correctly, e.g 𝔊 (G) or 𝔖 (S) [3].



(g) The word Constitution written with a Long S (ſ) [1].

(h) Use of a name location in French, *Porte de Namur* (Namur Gate), within an English document [2].

Fig. 1: Examples of historical newspaper documents.

tion process itself (e.g. digitization noise, image quality and OCR bias) [36,43,37,42,10,11,12].

Digitized historical documents do not only increase the availability of these resources but allow digital humanities researchers to search, structure and organize information located within the documents [44, 9]. For instance, researchers might use digitized documents to identify tangible keywords (i.e. people, places, events) but also more abstract, varied, and subtler concepts, such as themes and topics. Furthermore, digitized historical documents have allowed the use of natural language processing (NLP) tools, such as named entity recognition (NER) [10,11,12] and entity linking (EL) [36,37] for enriching automatically the documents. Which in turn have attracted their attention to other digital humanities researchers since they allow fostering further research towards finding patterns in historical documents regarding cultural changes, variations in gender bias across the historical periods, emerging technological trends, or transitions to new political ideas [53,27].

Despite the interest of digital humanities researchers in NLP and information retrieval (IR) tools, the creation of these for processing contemporary and historical documents has been disproportionate. For contemporary documents, in the last decade, the number of

tools has increased until the point where they have been generally adopted. However, this has not been the case for historical documents, due to certain characteristics, which make their processing particularly difficult. For instance, tools need to be able to deal with errors produced by OCR systems, to manage some specific vocabulary, and also to handle spelling variations with respect to modern standards. To ease the impact of OCR errors, one solution is to apply post-OCR correction [43], but while beneficial, this will still leave OCR errors in the text.

To illustrate and extend some of the aforementioned problems, we present in Figure 1 a collection of images representing historical newspapers or portions of them. As we can observe in Figure 1 (a-c), newspapers can have different templates but also face an unbalanced level of degradation. In the case of Figure 1 (c, d) we can observe a stamp that covers parts of the original text and makes illegible portions of it. Figure 1 (e) provides an example of a text containing a word that currently is spelled differently, which might difficult the match in contemporary knowledge bases. In Figure 1 (f-g), we present two fonts that can be difficult to process by an OCR system due to the geometry of certain characters, such as $\mathfrak{S}$ (S), $\mathfrak{P}$ (P), and $\mathit{\Gamma}$ (Long S). For instance, Figure 1 (g), the word "Con$\mathit{\Gamma}$titution" was recognized as "Conftitution" by an OCR system[1]. Finally, in Figure 1 (h), we present a document where we can notice a mix between French and English within a single document.

Apart from digitizing and recognizing the text, the processing of historical documents consists as well on extracting metadata from these documents. This metadata is used to index the key information inside documents to ease the navigation and retrieval process. Among all the possible key information available, named entities are of major significance as they allow structuring the document content [25], and correspond to key elements looked for in search engines [17]. These entities can represent aspects such as people, places, organizations, and events. Nonetheless, historical documents may contain duplicated and ambiguous information about named entities due to the heterogeneity and the mix of temporal references [52,28]. A disambiguation process is thus essential to distinguish named entities to be further utilized by search systems in digital libraries. For instance, "Bonaparte" can refer to several entities: the general "Napoleon Bonaparte"[2] or his son, "Napoleon François Joseph Charles Bonaparte"[3], but also a German band[4], to name a few.

Entity linking (EL) aims to recognize, disambiguate, and relate named entities to specific entries in a knowledge base. EL is a challenging task due to the fact that named entities may have multiple surface forms, for instance, in the case of a person an entity can be represented with their full or partial name, alias, honorifics, or alternate spellings [51]. Compared to contemporary data, few works in the state of the art have studied the EL task on historical documents [52,31,13,14,28,40,49] and OCR-processed documents [36].

In our previous work [47], we proposed a combination of a multilingual end-to-end entity linking method with several techniques to minimize the impact of issues frequently found in historical data. Our EL approach made use of entity embeddings, built from Wikipedia in multiple languages, along with a neural attention mechanism that analyzes context words and candidate entity embeddings to disambiguate mentions in historical documents. To reduce the impact of historical documents, we developed several modules to handle the multilingualism and errors related to OCR systems.

In this paper, we present MELHISSA, a Multilingual Entity Linking architecture for HIstorical preSS Articles, which extends our previous work on EL [47]. Specifically, we present an EL analysis on two recent historical datasets: CLEF HIPE 2020 [22] and NewsEye [26], that are composed of documents in English, Finnish, French, German, and Swedish. This deep analysis enabled us to improve our approach and achieve better results for both datasets and all languages.

This paper is organized as follows. We present an overview of EL approaches and a survey on historical data for the EL task in Section 2. Our multilingual approach is described in Section 3. Next, the CLEF HIPE 2020 and NewsEye datasets are described in Section 4. Then, the experimental setup is introduced in Section 5, while the results are presented in Section 6. We discuss the results in Section 7. And, finally, we provide the conclusions and some final comments in Section 8.

2 Entity Linking for Historical Data

Entity linking (EL) is an information extraction (IE) task that semantically enriches documents by identifying pieces of text that refer to entities, generally depicted as mention detection, and by matching each piece to an entry in a knowledge base (KB), also referred as entity disambiguation. Frequently, the detec-

[1] HIPE-data-v1.3-test-masked-bundle5-en.tsv#L56-L61

[2] https://www.wikidata.org/wiki/Q517

[3] https://www.wikidata.org/wiki/Q7723

[4] https://www.wikidata.org/wiki/Q892094

tion of mentions is delegated to an external named entity recognition (NER) system. In the state of the art of EL, the systems are either disambiguation systems [24,45], i.e. tools that perform only the matching of entities and consider the first task as an input, or end-to-end systems [34,18,15,31,40,49], i.e. tools that jointly perform both tasks, detecting and disambiguating the entities at the same time.

In the last year, new methods have been proposed for disambiguating entities and to solve specific issues, such as domain overfitting and context neglection. For instance, Onoe and Durrett [45] proposed a disambiguation system to overcome the risk of EL methods of overfitting to the domain (the genre of text or the particular distribution of entities), and, in consequence, to generalize effectively. The model does not rely on labeled entity linking data with a specific entity distribution was also proposed. The authors derive a large inventory of types from Wikipedia categories and use hyperlinked mentions in Wikipedia to distantly label data and train an entity typing model. With this domain-independent setting, their approach achieves strong results on the CoNLL dataset [50].

While most disambiguation systems employ entity representations embeddings bootstrapped from word embeddings to assess topic-level context compatibility, they also tend to neglect the context of the mention. A recent method, [16], injects latent entity type information into the entity embeddings based on the widely utilized pre-trained bidirectional encoder representations from Transformers (BERT) [20]. Then, it integrates a BERT-based entity similarity score into the local context model of a state-of-the-art model to better capture latent entity type information. This method significantly outperformed the state-of-the-art entity linking models on the standard benchmark (AIDA-CoNLL [30]).

With respect to end-to-end EL systems, these systems were initially defined for modern documents [18]. However, as time passed, researchers have been interested in end-to-end EL systems for historical documents [39]. Furthermore, the first end-to-end EL systems were focused on monolingual corpora and have gradually moved to cross-lingual and multilingual contexts. For example, a recent configuration, cross-lingual named entity linking (XEL), consists of analyzing documents and named entities in a language different from the one used in the knowledge base (KB). Several recent works proposed different XEL approaches: zero-shot transfer learning method by using a pivot language [48], a hybrid approach using language-agnostic features that combine existing lookup-based and neural candidate generation methods [54], and the use of multilingual word embeddings to disambiguate mentions across languages [37].

Another work [55] proposed a new approach to assess the problems faced by their previous entity candidate generation methods [54] for low-resource XEL. They reduce the disconnection between entity mentions and KB entries by introducing mention-entity pairs into the training process to provide supervision. Also, their approach improves the robustness of the model to low-resource scenarios by adjusting their previous neural-based model.

Further, an end-to-end BERT-based system [15] was advocated for EL by casting as a token classification over the entire entity vocabulary (an entity vocabulary, in this case, would be of a considerably large amount, e.g. 700k). The authors showed on an entity linking benchmark that improved the entity representations over plain BERT and it outperformed EL architectures that optimized the tasks separately, while their system came second to the current state-of-the-art that performs mention detection and entity disambiguation jointly.

In Digital Humanities, EL systems dedicated to historical documents have also been explored [31,40,41, 49]. For instance, van Hooland *et al.* [31] evaluated three third-party entity extraction services through a comprehensive case study, based on the descriptive fields of the Smithsonian Cooper-Hewitt National Design Museum in New York. Ruiz and Poibeau [49] utilized DBpedia Spotlight tool[5] to disambiguate named entities on Bentham manuscripts[6]. Moreover, Munnelly and Lawless [41] investigated the accuracy and overall suitability of EL systems in 17$^{\text{th}}$ century depositions obtained during 1641 Irish Rebellion[7].

Most of the developed EL systems in Digital Humanities are monolingual. Several disambiguation systems have been studied by focusing on specific types of entities in historical documents, e.g. person and place names. Smith and Crane [52] investigated the identification and disambiguation of place names in the Perseus digital library. They concentrated on representing historical data in the humanities from Ancient Greece to 19$^{\text{th}}$ century America. In order to overcome the heterogeneous data and the mix of temporal references (e.g. places that changed their name through time), they proposed a method based on honorifics, generic geographic labels, and linguistic environments to recognize entities, while they made use of gazetteers, biographical information, and general linguistic knowledge to disam-

[5] https://www.dbpedia-spotlight.org/

[6] https://www.ucl.ac.uk/library/digital-collections/collections/bentham

[7] https://1641.tcd.ie/

biguate these entities. Other works [13,14] focused on author names in French literary criticism texts and scientific essays from the 19th and early 20th centuries. They proposed a graph-based method that leverages knowledge from different linked data sources to generate the list of candidates for each author mention. Then, it crawls data from other linked data sets using equivalence links and fuses graphs of homologous individuals into a non-redundant graph in order to select the best candidate.

End-to-end EL systems in Digital Humanities have also been developed [31,40,49]. Some works concentrated on developing features and rules for improving EL in a specific domain [28], others by efficiently utilizing entity types [52,13,14]. Furthermore, some researchers investigated the effect of the issues frequently found in historical documents on the task of EL [28, 36]. Most of the proposed systems were also monolingual. The work of Mosallam *et al.* [39] proposed a monolingual unsupervised method to recognize person names, locations, and organizations in digitized French journals of the National Library of France (Bibliothèque nationale de France) from the 19th century. Then, they used a French entity knowledge base along with a statistical contextual disambiguation approach. Interestingly, their method outperformed supervised approaches when trained on small amounts of annotated data. Huet et al.[32] also analyzed the French journal Le Monde archive, a collection of documents from 1944 until 1986 discussing different subjects (e.g. post-war period, end of colonialism, politics, sports, culture). The authors calculated a conditional distribution of the co-occurrence of mentions with their corresponding entities (Wikipedia article). Then, they linked these Wikipedia articles to YAGO [46] to recognize and disambiguate entities in the archive of Le Monde.

Heino *et al.* [28] investigated EL in a particular domain, the Second World War in Finland, using the reference datasets of WarSampo[8]. They proposed a ruled-based approach to disambiguate military units, places, and people in these datasets. Moreover, they investigated problems regarding the analysis and disambiguation of these entities in this kind of data while they proposed specific rules to overcome these issues.

Regarding the lack of resources in the context of Digital Humanities, there is a recently explored method [33] that tackles the low resource settings with hardly annotated data and domain-specific KBs. The approach proposes a domain-agnostic feedback-based annotation approach based on suggestions from the annotators of potential concepts and adaptive candidate ranking. The method proves to be improving the annotation process by 35% compared to annotating without interactive support.

EL in historical datasets relies on information such as names or locations that are both non-unique and prone to enumeration and transcription errors. These errors make it impossible to find the correct match with certainty. Another recent paper [7] brings forward a fully automated probabilistic method for linking historical datasets that enable researchers to create samples at the frontier of minimizing type I (false positives) and type II (false negatives) errors, by utilizing the expectation-maximization (EM) algorithm. The authors study the method to link historical population censuses in the US and Norway and use these samples to estimate measures of intergenerational occupational mobility.

The impact of OCR errors on EL systems, to our knowledge, has rarely been analyzed or alleviated in previous research. Thus, the ability of EL to handle noisy inputs continuous to be an open question. Nevertheless, Linhares Pontes *et al.* [36], reported that EL systems for contemporary documents can see their performance decreasing around 20% when OCR errors, at the character and word levels, reach rates of 5% and 15% respectively.

Differently from previous works, we propose a multilingual end-to-end approach to link entities mentioned in historical documents to a KB containing several techniques to reduce the impact of the issues generated by the historical data issues, e.g. multilingualism, grammatical errors generated by OCR engines, linguistic historical word variations. We continue in the next section by detailing our approach.

3 Multilingual End-to-end Entity Linking

As aforementioned, historical documents present particular characteristics that make challenging the use of EL. In the following subsections, we describe the methods and techniques we developed for creating *MELHISSA*, our EL system that addresses these challenges.

3.1 Building Resources

The main component of an EL system is its knowledge base (KB) which allows the storage of the full list of entities use as reference. Moreover, modern KBs are rich enough to deal with additional tasks such as extraction of supplementary contexts or surface names, disambiguation of cases, or linking of entities with a particular website entry. A well-known set of publicly

[8] https://seco.cs.aalto.fi/projects/sotasampo/en/

available KBs are Wikipedia[9], Wikidata[10], and DBpedia [35]. Here, we briefly describe these KBs.

Wikipedia is a multilingual encyclopedia that includes more than 300 languages but only near to 70 languages have more than 100,000 articles. It is a widely used KB as a source of information for EL systems but also for building datasets. Multiple research studies, e.g. [24,34,38], make use of the English Wikipedia to train their models and disambiguate entity mentions. However, it has been also used to study the matching of mentions to Wikipedia articles based exclusively on their cultural heritage as well as for the disambiguation of mentions found in historical documents [8].

Wikidata is a KB created by the Wikimedia Foundation[11]. Its main purpose is to store user-generated data from the various projects supported by Wikimedia. Wikidata is widely used as a standard reference for entities, in the context of digital humanities, Wikidata has been used to annotate CLEF HIPE 2020 and NewsEye, two EL datasets of historical documents.

DBpedia is a KB that categorizes data from different Wikimedia projects, like Wikipedia and Wikidata. Furthermore, it associates this information to other KBs such as YAGO [46] and GeoNames[12]. It has been used in different projects related to EL [23,31,19,40]. For instance, De Wilde [19] used it for linking locations in a historical newspaper corpus. And, Munnelly and Lawless [40] utilized DBpedia for annotating historical legal documents.

We built our own KB mainly based on Wikipedia. In order to cover a large number of languages and long-tail entities, we make use of the Wikipedia versions of our target languages, e.g. French, German, Finnish, and Swedish, as well as the English Wikipedia. Our idea behind this strategy is that despite the richness and coverage of the English Wikipedia, in some cases other versions of Wikipedia might contain information that is only found in a specific language. This situation is less frequent for popular entities but common when dealing with long-tail entities. For instance, *Maurice Maréchal*, a journalist and founder of the French newspaper *Le Canard enchaîné*, has entries only in the French and Esperanto Wikipedias[13].

3.2 Probabilistic Table Entity−Map

In order to provide relevant candidates for mentions, we extracted the last version of the Wikipedia dump and analyzed the Wikipedia pages. We collected all hyperlinks presented in these pages to map the Wikipedia pages to their surface variation names represented in the hyperlinks. For instance, Figure 2 shows examples extracted from the English Wikipedia. While most mentions (in color blue) has the same surface representation as their links to the Wikipedia pages (e.g. "Association football", "1904 Summer Olympics", "St. Louis", "Canada", and "Ontario"), the mention "United States" represents the Olympic and Paralympic committee of United States which has a shorter surface representation of its mention. The mention United States can also represent the country in Figure 2(b). Finally, Figure 2(c) shows an example where the mention "United States of America" is longer than its entity ("United States").

From these maps, we calculate the probability of an entity (Wikipedia page) e to be related to a mention (surface representation) m:

$$p(e|m) = \frac{|m \mapsto e|}{|m|} \quad (1)$$

where $|m \mapsto e|$ is the number of times that mention m refers e within Wikipedia and $|m|$ is the total number of occurrences of the mention m in the Wikipedia dump. From this probabilistic table, it is possible to find which are the top entities that a mention span refers to. For instance, the mention "United States" has a probability of 95.9% to be related to the entity "United States" and 1×10^{-6} to the entity "United States Olympic & Paralympic Committee".

3.3 Entity embeddings

We create entity embeddings for each language, in the same manner as in [24], by generating two conditional probability distributions:

– the *positive probability distribution* is an approximation based on the word-entity co-occurrence counts, i.e. which words appear in the context of an entity. These counts were obtained from the entity

[9] https://www.wikipedia.org
[10] https://www.wikidata.org
[11] https://www.wikimedia.org
[12] http://www.geonames.org
[13] https://fr.wikipedia.org/wiki/Maurice_Mar%C3%A9chal_(journaliste)

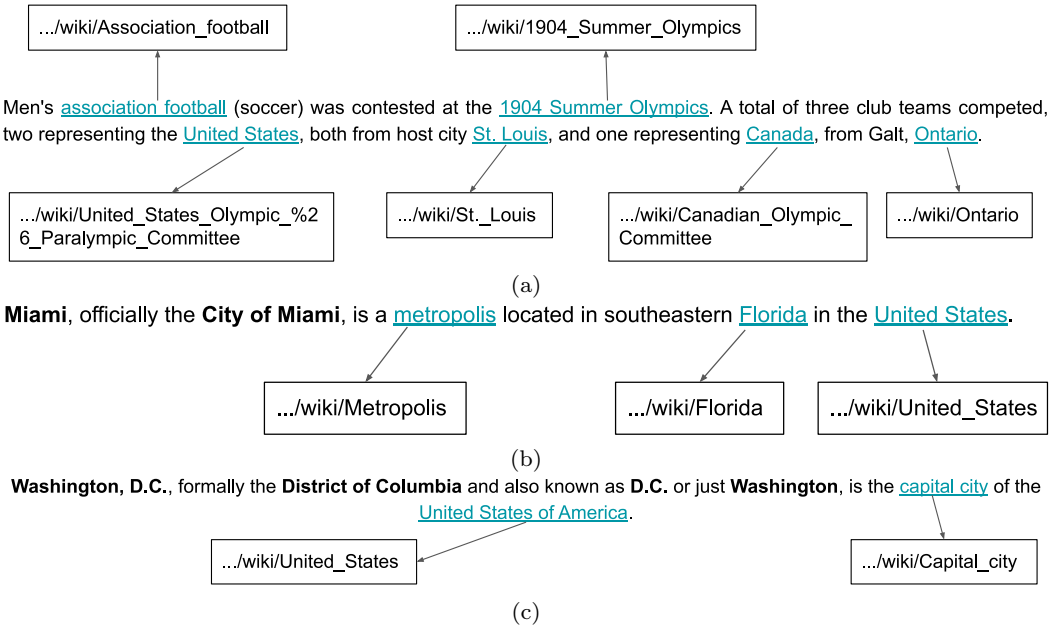


Fig. 2: Examples of mentions and their links to Wikipedia pages. Sentences extracted from the Wikipedia pages: `https://en.wikipedia.org/wiki/Football_at_the_1904_Summer_Olympics`, `https://en.wikipedia.org/wiki/Miami`, and `https://en.wikipedia.org/wiki/Washington,_D.C.`.

Wikipedia pages, and from the surrounding context of the entity in the corpus, by utilizing a fixed-length window.
- the *negative probability distribution* is calculated by random sampling of context windows that were unrelated to a specific entity.

These probability distributions were utilized with the purpose of changing the alignment of the word embeddings with respect to an entity embedding. While the *positive probability distribution* should approach the embeddings of the co-occurring words with the entity embedding, *the negative probability distribution* should distance the word embeddings that affiliated or related to an entity.

In order to prevent bias and low generalization, we create these word embeddings by not relying or depending on the dataset. In the case where an entity does not have entity embeddings, the EL system will propose a NIL.

3.4 Entity disambiguation

To disambiguate entities, we make use of a neural end-to-end model based on a BiLSTM and different types of embeddings. Specifically, the architecture follows the original model proposed by Kolitsas et al. [34] and depicted in Figure 3.

The reason for using this architecture is that it performs both entity linking and entity disambiguation. Therefore, the use of the systems is simplified and less prone to the propagation of errors. Moreover, this neural architecture does not need complex feature engineering. Thus, it is easy to adapt to multiple languages other than English.

For recognizing all entity mentions in a document, we, as Kolitsas *et al.* did, make use of an empirical probabilistic table entity−map, which has been described previously in Section 3.2.

Our end-to-end EL model starts by encoding every input token into dense representations. This is done by concatenating word and character embeddings which then are fed into a Bidirectional Long Short Term Memory (BiLSTM) [29] network. The BiLSTM network projects the document's mentions into a shared dimensional space, which has the same size as embeddings generated for the entities. The entity embeddings is a collection of fixed continuous entity representations generated using the approach described by Ganea and Hofmann [24], and aforementioned in Subsection 3.3.

To analyze long context dependencies of mentions, we make use of the attention mechanism defined by Ganea and Hofmann [24]. Specifically, this mechanism provides one context embedding per mention. This context is based on surrounding context words that are related to at least one of the candidate entities.

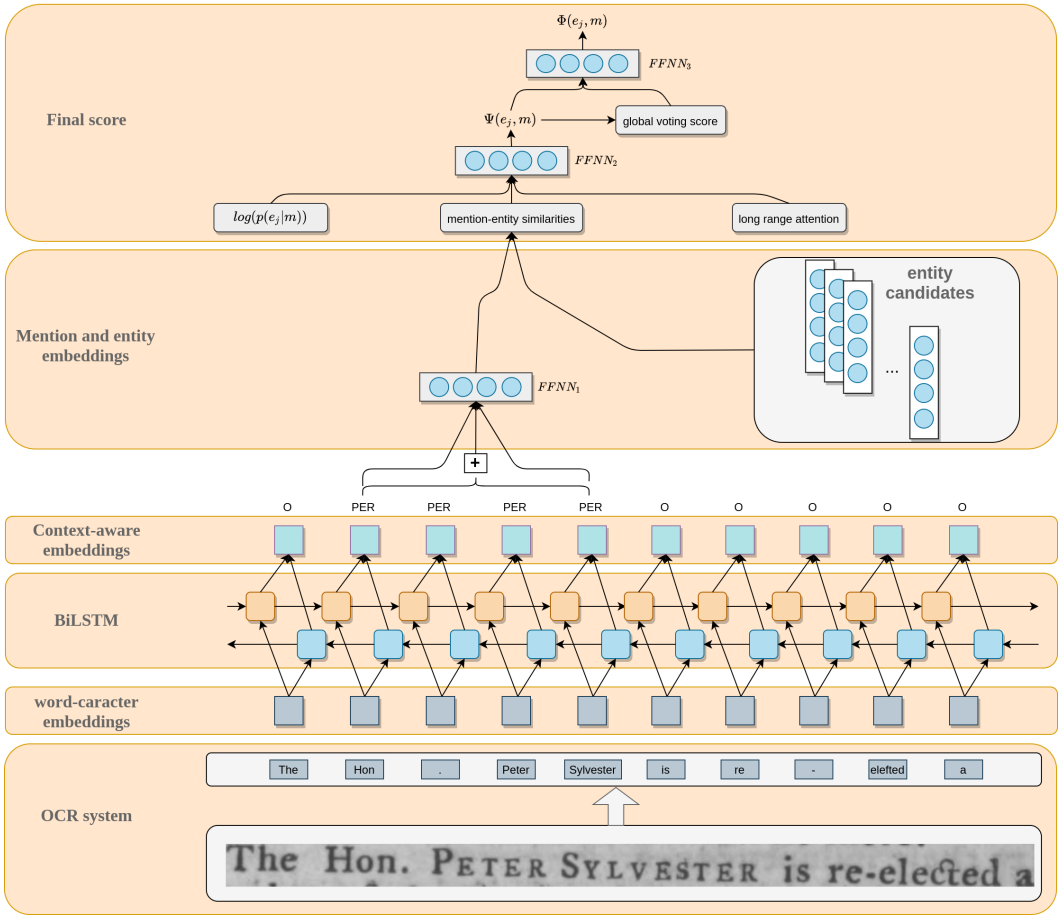


Fig. 3: Our global model architecture shown for the mention *Hon. Peter Sylvester* (from dev data of CLEF HIPE 2020). The final score is used for both the mention linking and entity disambiguation decisions.

For each mention, the final score is determined by combining the $\log p(e|m)$, similarity between a mention and candidate entity, and the long-range context attention for this mention. Finally, the consistency between disambiguated entities within a document is promoted by a top layer in the neural network.

To minimize the impact of historical data issues, we proposed two techniques. Section 3.5 presents a match correction that alleviates OCR-related issues, while Section 3.6 proposes a method to deal with multilingualism. Furthermore, we propose in Section 3.7 a post-processing filter to increase the performance of our EL systems.

3.5 Match Corrections

Multiple EL approaches, including the one used in this work, rely on the matching of entities and candidates using a probability table. If an entity is not listed in the probability table, the EL system cannot disambiguate it and, therefore, cannot propose candidates. In historical documents, not matching entities is a frequent problem, due to their inherent nature and processing, as explained in Section 1.

Multiple heuristics are used to analyze several surface name variations in order o increase the matching of entities in the probability table. These variations can deal with the casing (lower and upper, capitalization), with the concatenation of surrounding words, the removal of stopwords, or the transliteration to Latin characters some special characters like accentuated letters.

Previous heuristics do not prevent to miss matches. In that case, weighted Levenshtein distance is used to overcome more complex cases like transcription errors or spelling mistakes. We followed the idea exposed in [43] by using a mapping of OCR error calculated on historical documents that helps in identifying common OCR mistakes (*e.g* confusion between 'e' and 'c'). In this work, the average percentage mapping of OCR errors that are described in [43] is used to set up some weights in the Levenshtein distance.

3.6 Multilingualism

One of the biggest challenges in EL is the link of a mention, for which a KB has no entry. Either because it is known differently in specific languages or because the KB is not large enough to cover the topic.

For instance, in Figure 1(h), we presented the case of an English document making reference to the Namur Gate using its French name, "Porte de Namur". While the English Wikipedia contains an entry regarding the Namur Gate, only in the French Wikipedia it is known as "Porte de Namur". This makes it impossible to find, on occasions, the correct entry to which a mention should be linked.

To solve this issue, in the work, we combine the probability tables generated by different languages, in order to create one multilingual probability table. In this way, the EL system can match mentions' surface names with entries in multiple languages.

3.7 Filtering

To improve the accuracy of the candidates provided by the EL systems, we use a post-processing filter based on heuristics and data provided by Wikidata and DBpedia.[14] The goals of the filter are to: 1) Remove candidates which are improbable such as disambiguation pages or people born after the document publication; 2) Fix redirection page issues; 3) Reorder the candidates based on their DBpedia type classification or how similar the candidate label is to the named entity to link.

The filter consists of four main steps, which are described as follows and presented graphically in Figure 4.

The first step resides on querying Wikidata for five elements: *redirection_page* (boolean), *disambiguation_page* (boolean), *label* (string), *alternative_labels* (collection of strings) and *entry_year* (numeric). The former element helps us to find the correct page from which to extract the other elements.[15] For instance, the ID Q63832446 redirects automatically to Q4182026.[16] The second element, *disambiguation_page*, indicates whether we need to remove the candidate ID as a link cannot refer to an ambiguous entry, such as Moon (Q2432366)[17]. If the candidate ID is not a disambiguation page, we request Wikidata the label, and the alternative labels, if exist, associated with the entry in the language of analysis. For instance, the English entry of *Namur Gate* has as alternative label *Naamsepoort*.[18] Furthermore, we query Wikidata the year in which the entry was conceived. For example, in the case of a person, it would be their birth year, while for a book (product), the year in which was published, or for a country (location), their inception date.

The second step filters the candidate ID based on their entry year and the publication year. In the case that either the entry or the publication is not associated with a year, the step is skipped. Furthermore, it is possible to specify which mention types are filtered by year.

The third step relies on querying DBpedia whether the candidate ID exists in their knowledge base and whether it is associated with specific categories defined for each mention type. If DBpedia does not contain the candidate ID or it does not link it to the specific categories, we request the same information to the DBpedia Chapters. We show in Table 1 the DBpedia types associated for each mention type. The categories associated with each mention type were manually defined. From this step, we generate three types of candidates:

- Top: These IDs were considered in DBpedia to represent the mention type.
- Middle: This type is for IDs for which it was impossible to retrieve information from either DBpedia or DBpedia Chapters. These IDs can be considered as part of bottom candidates depending on the filter configuration.
- Bottom: It represents those candidates that were found in DBpedia, but according to their classification, they do not match the mention type.

Finally, the fourth step of the filter consists of sorting the three types of candidates defined in the previous step. The candidates are sorted based on incremental edit distances between the label (or alternative labels)[19] and the mention found in the text analyzed. The system breaks ties using the ordering in which the candidates were presented by the EL system. Once all the candidates have been sorted, they are printed as follows: 1) Top candidates 2) Middle candidates 3) NIL and 4) Bottom candidates. The addition of a NIL before the Bottom candidates is due to the fact, that we consider it less probable that a mention is linked to an ID that does not match DBpedia classification. Furthermore, if no top candidates were found, it might be probable that the mention should be linked to NIL.

We present in Figure 5 an example of the filtering process for the named entity "Great Britain" found in

[14] Code available at: `https://github.com/EMBEDDIA/NEL_Filter`

[15] Most of the redirections occur when two entries in Wikidata were merged. See: `https://www.wikidata.org/wiki/Help:Redirects`

[16] `https://www.wikidata.org/wiki/Q63832446`

[17] `https://www.wikidata.org/wiki/Q2432366`

[18] `https://www.wikidata.org/wiki/Q3399071`

[19] We take the string that produces the shortest distance.

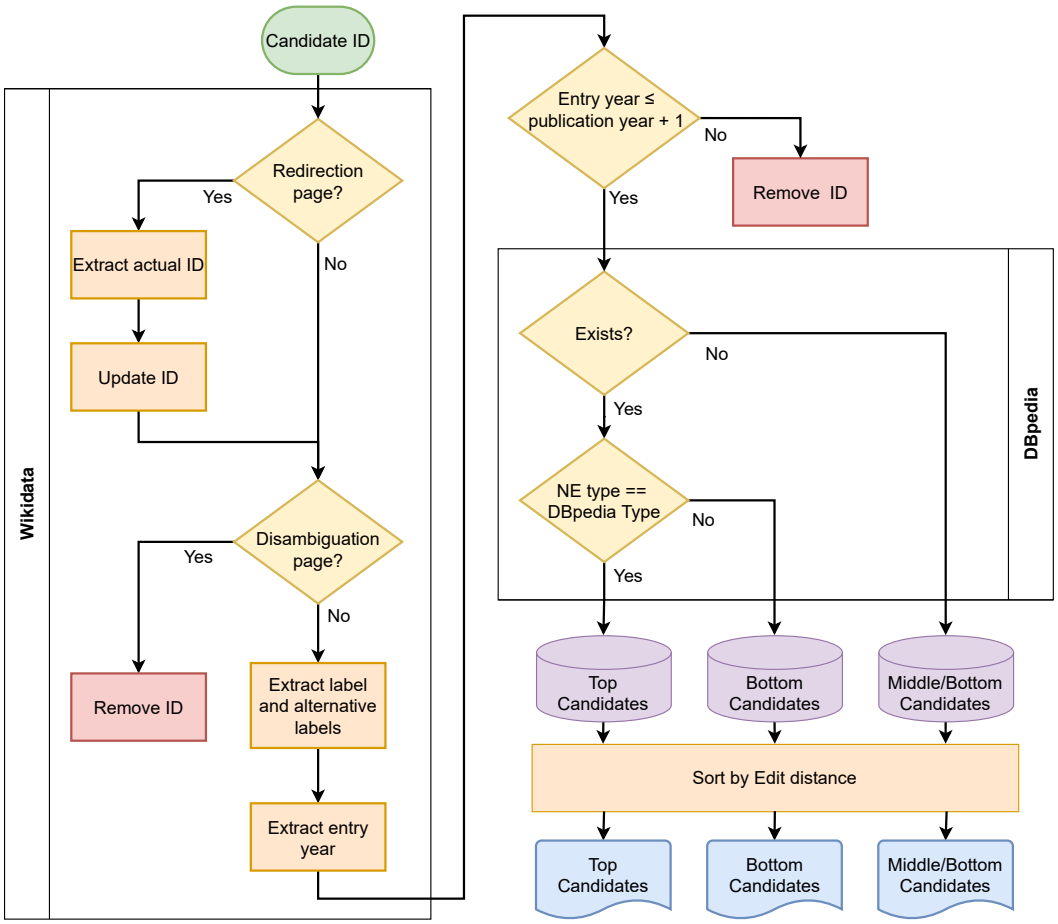


Fig. 4: Flowchart of the filtering module.

Table 1: Relation between each type of mentions and their associated DBpedia types.

Mention Type	**Associated DBpedia Type**
Location (LOC)	dbo:Location, dbo:Place, dbo:Settlement, dbo:Region, dbo:Building, dbo:Village, umbel-rc:Country, yago:YagoGeoEntity
Organization (ORG)	dbo:Organisation, umbel-rc:Business, dbc:Supraorganizations, yago:YagoGeoEntity
Person (PER)	foaf:Person, dbo:Person, dbo:Agent, dul:SocialPerson
Product (PRO)	dbo:Work, dbo:Newspaper, umbel-rc:Business, schema:CreativeWork, yago:TradeName106845599, yago:Product104007894

a publication of 1868. As we can observe in Figure 5, some of the candidate IDs make reference to entities that started to exist long after the publication of the document. As well, not all the proposed candidates are considered as locations in DBpedia.

This filter architecture differs from the one proposed in our previous work [47], on the fact that the labels are obtained from Wikidata, instead of DBpedia. But also that we query for the DBpedia types to all the existing DBpedia services, i.e. this includes DBpedia and DBpedia Chapters, instead of just a subset of them. As well, we can filter different types of mentions and not just those related to people. Furthermore, we fix redirection pages, which were ignored previously, and we sort middle and bottom candidates according to their edit distances.

4 Historical Datasets

While EL on contemporary datasets can take advantage of abundance of resources and tools [24,45,34,18,15,31,40,49], digitized and historical documents lack annotated resources [31,40,41,49,28,36]. Moreover, contemporary datasets and resources are, generally, not

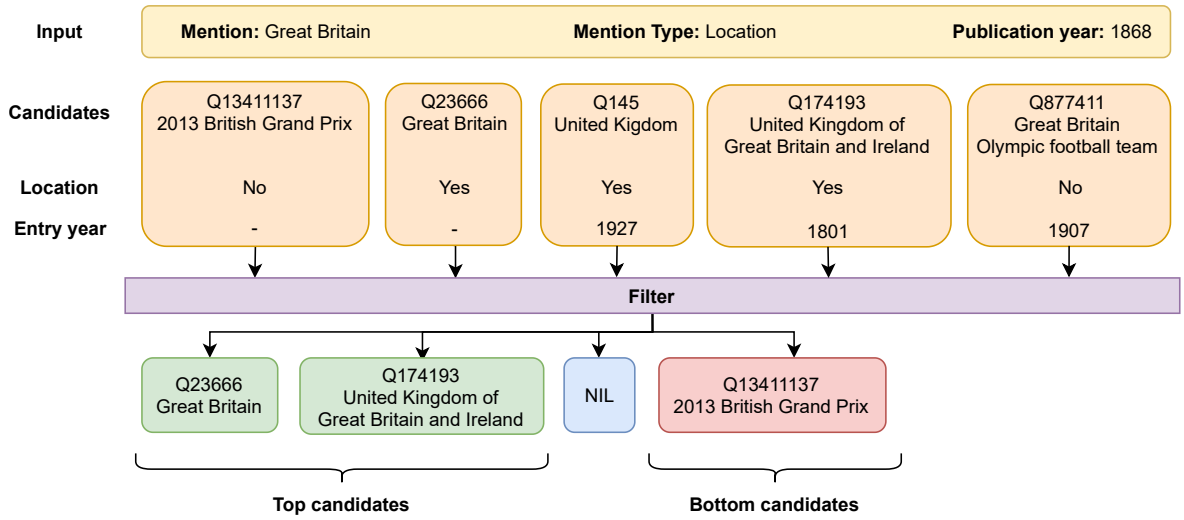


Fig. 5: Example of the filter application for the mention *Great Britain* in a 1868 publication.

suitable for building accurate systems for them to be applied to historical datasets due to several issues, i.e., the variations in orthographic and grammatical rules, historical word variations, and also the fact that names of persons, organizations, or places could have significantly changed over time [28,36].

To the best of our knowledge, there are few publicly available corpora in the literature with manually annotated entities on historical documents [22,10,11]. Most of the EL datasets contain contemporary documents [24,45,34] lacking the distinctive features found in historical documents.

In this paper, we focus on two datasets that contain historical documents in English, Finnish, French, German, and Swedish.

The first corpus was produced for the CLEF HIPE 2020 challenge[20] [21]. This corpus is composed of articles published between 1738 and 2019 in Swiss, Luxembourgish, and American newspapers. To build the corpus, the organizers randomly sampled articles from different newspapers according to predefined decades. For each newspaper, articles were randomly sampled among articles that belong to the first years of a set of predefined decades covering the lifespan of the newspaper, and have a title, have more than 50 characters, and belong to any page. It was manually annotated by native speakers according to HIPE annotation guidelines [21].

The second corpus is the NewsEye dataset[21] [26] which is composed of a collection of annotated historical newspapers in French, German, Finnish, and Swedish. These newspapers were collected by the national libraries of France[22] (BnF), with documents from 1854 to 1946, Austria[23] (ONB) with documents from 1864 to 1933, and Finland[24] (NLF), with Finnish and Swedish documents from respectively 1852 and 1848 to 1918.

Tables 2 and 3 describe the number of mentions by a period of time for the CLEF HIPE 2020 and the NewsEye datasets respectively. The named entities from both datasets are classified according to their type and, when possible, linked to their Wikidata ID. The entities that do not exist in the Wikidata KB are linked to NIL entries.

5 Experimental Settings

For all the languages, we utilize the multilingual pre-trained model MUSE [25]. Specifically, this pre-trained is used for the entity embeddings and disambiguation model. The MUSE word embeddings are 300-sized while the character embeddings are 50-sized.

As CLEF HIPE 2020 does not provide a training dataset for English, we make use of the contemporary corpus AIDA [30] for training purposes. Then the generated model is validated on the CLEF HIPE 2020 corpus.

Based on the statistical analysis of the training data, we defined the weighted Levenshtein distance ratio of 0.9, 0.94, 0.85, 0.89, and 0.82 for the languages German, English, Finnish, French, and Swedish, respectively, to search for other mentions in the probability table if this

[20] https://impresso.github.io/CLEF-HIPE-2020/
[21] https://zenodo.org/record/4573313#.YH79nnUzY5k
[22] https://www.bnf.fr
[23] https://www.onb.ac.at
[24] https://www.kansalliskirjasto.fi
[25] https://github.com/facebookresearch/MUSE

Table 2: Number of mentions by period of time in the CLEF HIPE dataset.

Splits	German				English					French					
	1750-1800	1800-1850	1850-1900	1900-1950	1750-1800	1800-1850	1850-1900	1900-1950	1950-2000	1750-1800	1800-1850	1850-1900	1900-1950	1950-2000	>2000
Train															
ORG	8	56	70	74	–	–	–	–	–	10	38	50	116	88	14
LOC	12	84	105	111	–	–	–	–	–	15	57	75	174	132	21
PERS	16	112	140	148	–	–	–	–	–	20	76	100	232	176	28
PROD	20	140	175	185	–	–	–	–	–	25	95	125	290	220	35
TIME	24	168	210	222	–	–	–	–	–	30	114	150	348	264	42
Dev															
ORG	6	26	22	26	10	54	26	44	26	4	14	8	32	24	4
LOC	9	39	33	39	15	81	39	66	39	6	21	12	48	36	6
PERS	12	52	44	52	20	108	52	88	52	8	28	16	64	48	8
PROD	15	65	55	65	25	135	65	110	65	10	35	20	80	60	10
TIME	18	78	66	78	30	162	78	132	78	12	42	24	96	72	12
Test															
ORG	2	20	34	42	6	32	14	30	10	6	16	16	26	16	6
LOC	3	30	51	63	9	48	21	45	15	9	24	24	39	24	9
PERS	4	40	68	84	12	64	28	60	20	12	32	32	52	32	12
PROD	5	50	85	105	15	80	35	75	25	15	40	40	65	40	15
TIME	6	60	102	126	18	96	42	90	30	18	48	48	78	48	18

Table 3: Number of mentions by period of time in the NewsEye dataset.

Splits	German		French			Finnish		Swedish		
	1850-1900	1900-1950	1800-1850	1850-1900	1900-1950	1850-1900	1900-1950	1800-1850	1850-1900	1900-1950
Train										
ORG	539	2,571	169	100	1,016	55	204	3	92	58
LOC	1,437	3,707	610	515	2,930	401	578	13	620	352
PERS	1,024	2,082	920	299	3,664	231	551	14	559	265
PROD	–	37	72	39	89	57	69	8	117	39
Dev										
ORG	9	114	18	45	71	11	26	1	11	5
LOC	72	191	64	45	226	22	75	8	68	72
PERS	31	118	64	24	187	11	66	4	59	21
PROD	–	4	2	6	3	2	10	2	2	13
Test										
ORG	21	116	24	9	184	15	6	–	11	3
LOC	157	340	161	67	369	42	42	8	90	42
PERS	122	123	155	36	272	51	40	21	87	34
PROD	1	2	6	9	6	3	4	1	11	3

mention does not have a corresponding entry in the probability table (Figure 6).

With respect to the post-processing filter, we query Wikidata, DBpedia and DBpedia Chapters using their respective SPARQL Query Service.[26]. Ten DBpedia Chapters are used: Catalan, Basque, Greek, Indonesian, Dutch, French, German, Japanese, Korean and Spanish. Furthermore, we explore two edit distance metrics: RapidFuzz Weight Ratio[27] and Weighted Levenshtein Distance[28] with specific costs defined by [43]. As well, we explore whether candidates not found in DBpedia

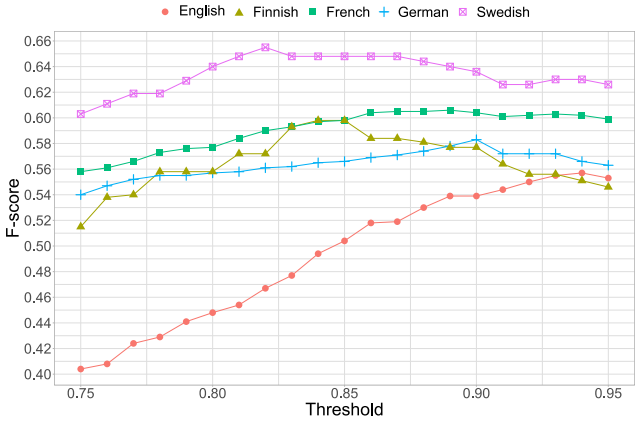


Fig. 6: F-score for different text distance thresholds to match mentions with OCR errors.

[26] Wikidata: `query.wikidata.org`, DBpedia: `https://dbpedia.org/sparql`, DBpedia Chapters: `https://wiki.dbpedia.org/join/chapters`

[27] `https://github.com/maxbachmann/rapidfuzz`

[28] `https://pypi.org/project/weighted-levenshtein/`

should be considered as middle candidates or bottom candidates. In total, we explore 18 different filters; their configuration is presented in Table 4.

Table 4: Filter configurations used in this work.

Filter	Edit Distance	Mentions to Filter by Date	Middle Candidates
1A	RapidFuzz W. Ratio	All	No
2A		None	
3A		Person	
4A		All	Yes
5A		None	
6A		Person	
1B	None	All	No
2B		None	
3B		Person	
4B		All	Yes
5B		None	
6B		Person	
1C	Weighted Levenshtein	All	No
2C		None	
3C		Person	
4C		All	Yes
5C		None	
6C		Person	

For evaluating our methods, we compute their F-score (F1) calculated for each language over the full corpus (micro-averaging).[29] Specifically, the F-score is defined as the harmonic mean between precision and recall. Where precision is the fraction of correctly linked entity mentions that are generated by a system. And recall takes into account all entity mentions that should be linked and determines how many correct linked entity mentions are with regard to total entity mentions that should be linked. It should be indicated, that not all the mentions in the corpora have a corresponding entry in Wikidata, for instance, ambiguous names such as Peter or Thomas. For these cases, the EL systems propose a NIL as a candidate, which is expected to match the NIL found in the gold standard too.

6 Results

We present in Table 5 and Table 6 the F-score obtained by each of the EL approaches detailed in Section 5 for the corpora CLEF HIPE 2020 and NewsEye respectively. As well, Table 5 and Table 6 contain the performance achieved by each post-processing filter applied to every base output generated by the EL systems.

From Table 5 and Table 6, we can notice that the match corrections, in general, improved the performance of the base EL candidates. Nonetheless, there are two languages, English CLEF HIPE 2020 and French NewsEye, where this approach reduced the performance of our EL systems. For the Swedish NewsEye dataset, only one configuration is negatively affected, i.e. when the match correction is coupled with a multilingual probability table.

Moreover, we can notice from Table 5 and Table 6 that in most cases the use of multilingual probability tables $p(e|m)$ reduced the performance of the EL systems. There are some partial exceptions, French CLEF HIPE 2020 and Swedish NewsEye, where multilingual probability tables $p(e|m)$ without match corrections performed better than monolingual probability tables without match corrections. Nevertheless, none of these two cases produces the best base EL performance in their respective languages.

As we can observe in Table 5 and Table 6, most of the EL configurations are benefited from the application of a post-processing filter. The only exception is for German CLEF HIPE 2020, where we used a monolingual probability table $p(e|m)$ and applied a match correction.

Although it is hard to observe in the first instance which filter is the best, we can notice certain patterns. In general, filters based on RapidFuzz Weight Ratio (filters A) generate the greatest number of top performances, especially in CLEF HIPE 2020 languages. Filters without re-ordering candidates based on edit distance (filters B), seem to generate the best performances for NewsEye Finnish and Swedish. Finally, filters based on a Weighted Levenshtein distance (filters C) produce the fewer number of best performances in both corpora, with some exceptions in CLEF HIPE 2020 English and NewsEye Finnish.

We can observe as well in Table 5 and Table 6, that in CLEF HIPE 2020 English and, NewsEye German and Finnish, it is better to filter all the mentions according to their date to get the best scores (filters 1). Nonetheless, filtering by date only the mentions of type person (filters 3) provide the best performance for CLEF HIPE 2020 German and French, as well as for NewsEye French and Swedish. This means that regardless of the corpus, mentions of type person are prone to be linked to entries that correspond to people born after the publication of the newspaper article.

As well, we can notice in Table 5 and Table 6, that for most datasets it is better not to use middle candidates (filters 1-3). The only exceptions are CLEF HIPE 2020 French and NewsEye Finnish, where it is better to separate mentions not found in DBpedia as middle candidates (filters 4-6).

[29] This has been done using the following tool: `https://github.com/impresso/CLEF-HIPE-2020-scorer`

Table 5: Analysis of the performance our EL approach with different hyperparameters on the CLEF HIPE 2020 dataset. Bold means the best performance on each configuration. Bold and italics means best performance on each language.

			F-score																	
				Filter																
Lang.	$p(e\|m)$	Match Cor.	Base	1A	2A	3A	4A	5A	6A	1B	2B	3B	4B	5B	6B	1C	2C	3C	4C	5C	6C
de	multi	False	0.514	0.524	0.533	0.535	0.523	0.532	0.534	0.519	0.525	0.527	0.518	0.524	0.527	0.527	0.536	**0.539**	0.527	0.535	0.538
de	multi	True	0.553	0.559	0.569	0.571	0.558	0.568	0.570	0.555	0.563	0.564	0.554	0.562	0.563	0.563	0.575	***0.577***	0.563	0.574	***0.577***
de	mono	False	0.532	0.526	0.529	**0.533**	0.524	0.527	0.531	0.526	0.528	0.532	0.524	0.527	0.530	0.525	0.528	0.532	0.523	0.527	0.53
de	mono	True	**0.572**	0.560	0.564	0.568	0.558	0.563	0.566	0.560	0.563	0.567	0.558	0.562	0.565	0.559	0.565	0.569	0.557	0.563	0.567
en	multi	False	0.569	**0.605**	0.596	0.598	0.603	0.592	0.594	0.598	0.589	0.592	0.596	0.585	0.587	0.598	0.589	0.592	0.596	0.585	0.587
en	multi	True	0.554	**0.611**	0.596	0.604	**0.611**	0.593	0.602	0.602	0.587	0.596	0.602	0.584	0.593	0.604	0.589	0.598	0.604	0.587	0.596
en	mono	False	0.603	**0.621**	**0.621**	**0.621**	**0.621**	0.619	0.619	0.619	0.619	0.619	0.619	0.616	0.616	0.607	0.607	0.607	0.607	0.605	0.605
en	mono	True	0.596	***0.636***	0.629	***0.636***	0.634	0.625	0.631	0.634	0.627	0.634	0.631	0.622	0.629	0.622	0.616	0.622	0.62	0.611	0.618
fr	multi	False	0.601	0.599	0.608	0.611	0.602	0.611	**0.614**	0.595	0.603	0.607	0.598	0.606	0.61	0.595	0.601	0.606	0.597	0.604	0.609
fr	multi	True	0.624	0.628	0.635	0.64	0.632	0.637	***0.643***	0.622	0.627	0.633	0.626	0.629	0.636	0.621	0.626	0.632	0.625	0.628	0.636
fr	mono	False	0.594	0.593	0.601	0.604	0.596	0.603	**0.607**	0.589	0.596	0.600	0.592	0.599	0.603	0.589	0.596	0.6	0.592	0.599	0.603
fr	mono	True	0.629	0.629	0.636	0.639	0.633	0.638	***0.643***	0.623	0.629	0.634	0.627	0.631	0.637	0.625	0.631	0.636	0.629	0.633	0.639

Table 6: Analysis of the performance our EL approach with different hyperparameters on the NewsEye dataset. Bold means the best performance on each configuration. Bold and italics means best performance on each language.

			F-score																	
				Filter																
Lang.	$p(e\|m)$	Match Cor.	Base	1A	2A	3A	4A	5A	6A	1B	2B	3B	4B	5B	6B	1C	2C	3C	4C	5C	6C
de	multi	False	0.538	**0.576**	0.567	0.574	0.574	0.564	0.571	0.574	0.563	0.57	0.571	0.561	0.568	0.574	0.565	0.573	0.571	0.563	0.570
de	multi	True	0.544	**0.590**	0.576	0.583	0.583	0.569	0.576	0.584	0.574	0.581	0.577	0.567	0.574	0.588	0.575	0.582	0.581	0.568	0.575
de	mono	False	0.548	0.585	0.574	0.583	0.583	0.57	0.581	**0.586**	0.574	0.583	0.584	0.570	0.581	0.582	0.571	0.581	0.579	0.568	0.578
de	mono	True	0.554	***0.597***	0.580	0.590	0.590	0.573	0.583	0.594	0.581	0.591	0.588	0.575	0.584	0.593	0.578	0.588	0.586	0.571	0.581
fi	multi	False	0.582	0.597	0.582	0.587	0.603	0.587	0.592	0.603	0.587	0.592	**0.608**	0.592	0.597	0.603	0.582	0.587	**0.608**	0.587	0.592
fi	multi	True	0.610	0.635	0.630	0.635	0.635	0.63	0.635	0.635	0.630	0.635	0.635	0.630	0.635	**0.640**	0.630	0.635	**0.640**	0.630	0.635
fi	mono	False	0.621	0.633	0.626	0.626	0.633	0.626	0.626	**0.643**	0.636	0.6360	**0.643**	0.636	0.636	0.638	0.626	0.626	0.638	0.626	0.626
fi	mono	True	0.645	0.663	0.660	0.660	0.663	0.650	0.660	***0.668***	0.665	0.665	***0.668***	0.655	0.665	***0.668***	0.660	0.660	***0.668***	0.650	0.660
fr	multi	False	0.569	0.600	0.600	**0.604**	0.599	0.598	0.603	0.597	0.598	0.602	0.596	0.596	0.600	0.594	0.596	0.599	0.593	0.593	0.597
fr	multi	True	0.543	0.597	0.596	**0.600**	0.587	0.584	0.59	0.594	0.594	0.597	0.584	0.582	0.588	0.591	0.591	0.594	0.581	0.58	0.584
fr	mono	False	0.582	0.613	0.613	0.617	0.609	0.610	0.613	0.614	0.614	***0.618***	0.609	0.611	0.614	0.611	0.612	0.615	0.607	0.609	0.612
fr	mono	True	0.559	0.614	0.613	**0.617**	0.599	0.600	0.604	0.614	0.613	**0.617**	0.599	0.600	0.604	0.612	0.612	0.615	0.598	0.599	0.603
sv	multi	False	0.590	0.616	0.612	0.625	0.609	0.612	0.619	0.619	0.616	**0.629**	0.612	0.616	0.622	0.616	0.612	0.625	0.609	0.612	0.619
sv	multi	True	0.580	0.642	0.629	0.651	0.635	0.619	0.645	0.645	0.632	***0.655***	0.638	0.622	0.648	0.642	0.629	0.651	0.635	0.619	0.645
sv	mono	False	0.574	0.594	0.597	**0.600**	0.587	0.591	0.594	0.594	0.597	**0.600**	0.587	0.591	0.594	0.594	0.597	**0.600**	0.587	0.591	0.594
sv	mono	True	0.597	0.633	0.633	**0.639**	0.626	0.617	0.633	0.633	0.633	**0.639**	0.626	0.617	0.633	0.633	0.633	**0.639**	0.626	0.617	0.633

Table 7: Analysis of the performance, in terms of F-score@5, regarding our EL approach with different hyperparameters on the CLEF HIPE 2020 dataset. Bold means the best performance on each configuration. Bold and italics means best performance on each language.

			F-score																	
				Filter																
Lang.	$p(e\|m)$	Match Cor.	Base	1A	2A	3A	4A	5A	6A	1B	2B	3B	4B	5B	6B	1C	2C	3C	4C	5C	6C
de	multi	False	0.593	0.603	0.618	0.618	0.603	0.618	0.618	0.603	0.617	0.617	0.603	0.617	0.617	0.603	**0.62**	**0.62**	0.603	**0.62**	**0.62**
de	multi	True	0.637	0.649	0.667	0.667	0.649	0.667	0.667	0.649	0.666	0.666	0.649	0.666	0.666	0.649	***0.669***	***0.669***	0.649	***0.669***	***0.669***
de	mono	False	0.584	0.593	**0.609**	**0.609**	0.593	**0.609**	**0.609**	0.593	**0.609**	**0.609**	0.593	**0.609**	**0.609**	0.593	**0.609**	**0.609**	0.593	**0.609**	**0.609**
de	mono	True	0.628	0.637	**0.655**	**0.655**	0.637	**0.655**	**0.655**	0.637	**0.655**	**0.655**	0.637	**0.655**	**0.655**	0.637	**0.655**	**0.655**	0.637	**0.655**	**0.655**
en	multi	False	0.62	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**	**0.693**
en	multi	True	0.611	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**	**0.708**
en	mono	False	0.634	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**	**0.702**
en	mono	True	0.634	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***	***0.719***
fr	multi	False	0.651	0.673	**0.687**	0.686	0.673	**0.687**	0.686	0.672	**0.687**	**0.687**	0.672	**0.687**	**0.687**	0.672	0.686	0.685	0.672	0.686	0.685
fr	multi	True	0.679	0.715	0.73	0.73	0.715	0.73	0.73	0.715	**0.731**	0.73	0.715	**0.731**	0.73	0.715	0.729	0.729	0.715	0.729	0.729
fr	mono	False	0.646	0.666	0.678	0.679	0.666	0.679	0.679	0.666	0.679	**0.68**	0.666	**0.68**	**0.68**	0.666	0.678	0.678	0.666	0.676	0.676
fr	mono	True	0.686	0.718	0.73	***0.732***	0.718	***0.732***	***0.732***	0.718	0.73	***0.732***	0.718	***0.732***	***0.732***	0.717	0.73	0.73	0.717	0.729	0.729

In Table 7 and Table 8, we present the performance of the EL systems calculating the F-score@5.[30] As we can observe in Table 7 and Table 8, the increment in the performance for the base EL systems when evaluating @1 and @5, indicates that in multiple cases the correct entry for a mention is found among the top 5 candidates.

Moreover, we can notice in Table 7 and Table 8 that by applying a post-processing filter, we can still increase the performance. For instance, in NewsEye French we can have an increment of up to 30%. As well, by measuring the F-score@5, it is easier to observe certain patterns among the filters, such as filtering all mentions by date tends to be worse than just filtering by date mentions of type person.

[30] This means, that a mention will be correctly linked if the entry is among the top 5 candidates.

Table 8: Analysis of the performance, in terms of F-score@5, regarding our EL approach with different hyperparameters on the NewsEye dataset. Bold means the best performance on each configuration. Bold and italics means best performance on each language.

			F-score																	
				Filter																
Lang.	$p(e\|m)$	Match Cor.	Base	1A	2A	3A	4A	5A	6A	1B	2B	3B	4B	5B	6B	1C	2C	3C	4C	5C	6C
de	multi	False	0.583	0.752	0.756	0.757	0.752	0.756	0.757	0.752	0.754	0.757	0.752	0.754	0.757	0.753	0.757	**0.758**	0.752	0.756	0.757
de	multi	True	0.593	0.785	0.789	0.79	0.785	0.789	0.79	0.785	0.787	0.79	0.785	0.787	0.79	0.786	0.79	**0.791**	0.785	0.789	0.79
de	mono	False	0.589	0.758	0.763	0.763	0.758	0.763	0.763	0.758	0.763	0.763	0.758	0.763	0.763	0.759	**0.764**	**0.764**	0.758	0.763	0.763
de	mono	True	0.598	0.788	0.792	0.793	0.788	0.792	0.793	0.788	0.792	0.793	0.788	0.792	0.793	0.789	0.793	***0.794***	0.788	0.792	0.793
fi	multi	False	0.623	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**	**0.653**
fi	multi	True	0.655	0.69	***0.695***	***0.695***	0.69	***0.695***	***0.695***	0.69	***0.695***	***0.695***	0.69	***0.695***	***0.695***	0.69	***0.695***	***0.695***	0.69	***0.695***	***0.695***
fi	mono	False	0.636	**0.658**	0.656	0.656	**0.658**	0.656	0.656	**0.658**	0.656	0.656	**0.658**	0.656	0.656	**0.658**	0.656	0.656	**0.658**	0.656	0.656
fi	mono	True	0.665	0.694	***0.695***	***0.695***	0.694	***0.695***	***0.695***	0.694	***0.695***	***0.695***	0.694	***0.695***	***0.695***	0.694	***0.695***	***0.695***	0.694	***0.695***	***0.695***
fr	multi	False	0.608	0.719	**0.725**	**0.725**	0.719	**0.725**	**0.725**	0.719	**0.725**	**0.725**	0.717	0.724	0.724	0.719	**0.725**	**0.725**	0.717	0.724	0.724
fr	multi	True	0.581	0.745	***0.752***	***0.752***	0.745	***0.752***	***0.752***	0.745	***0.752***	***0.752***	0.744	0.75	0.75	0.745	***0.752***	***0.752***	0.744	0.75	0.75
fr	mono	False	0.61	0.712	0.719	0.719	0.712	0.719	0.719	0.712	0.719	0.719	0.712	0.719	0.719	0.714	**0.721**	**0.721**	0.712	0.719	0.719
fr	mono	True	0.59	0.741	0.748	0.748	0.741	0.748	0.748	0.741	0.748	0.748	0.741	0.748	0.748	0.742	**0.749**	**0.749**	0.741	0.748	0.748
sv	multi	False	0.638	0.678	**0.687**	**0.687**	0.678	**0.687**	**0.687**	0.678	**0.687**	**0.687**	0.678	**0.687**	**0.687**	0.678	**0.687**	**0.687**	0.678	**0.687**	**0.687**
sv	multi	True	0.629	0.72	***0.73***	***0.73***	0.72	***0.73***	***0.73***	0.72	***0.73***	***0.73***	0.72	***0.73***	***0.73***	0.72	***0.73***	***0.73***	0.72	***0.73***	***0.73***
sv	mono	False	0.62	0.656	0.666	0.666	0.656	0.666	0.666	0.659	**0.669**	**0.669**	0.659	**0.669**	**0.669**	0.659	**0.669**	**0.669**	0.659	**0.669**	**0.669**
sv	mono	True	0.643	0.701	0.711	0.711	0.701	0.711	0.711	0.705	**0.715**	**0.715**	0.705	**0.715**	**0.715**	0.705	**0.715**	**0.715**	0.705	**0.715**	**0.715**

Table 9: Amount of mentions of CLEF HIPE 2020 and NewsEye datasets that contain a corresponding entry in their language version of Wikipedia KB.

Splits	CLEF HIPE 2020						NewsEye							
	English		French		German		Finnish		French		German		Swedish	
	Total	KB	Total	KB	Total	KB	Total	KB	Total	KB	Total	KB	Total	KB
							Train							
Total	–	–	5,406	5,008	3,209	2,868	1,281	1166	3303	3127	695	632	1,447	1,241
ORG	–	–	554	509	247	208	92	81	352	291	165	149	72	50
LOC	–	–	3,333	3,244	2,009	1,922	837	801	1786	1,743	454	427	834	792
PERS	–	–	1,343	1,102	849	667	270	215	1057	997	73	53	395	312
PROD	–	–	149	128	77	58	82	69	108	96	3	3	146	87
							Dev							
Total	549	532	1,450	1,335	1,157	1,058	133	122	510	481	409	372	199	180
ORG	93	87	120	105	112	94	13	11	57	46	88	84	7	6
LOC	332	324	851	816	711	684	80	78	283	276	223	203	132	122
PERS	107	104	434	370	294	247	31	26	163	153	96	83	49	45
PROD	16	16	37	36	35	30	9	7	7	6	2	2	11	7
							Test							
Total	283	273	1,345	1,280	1,025	954	116	112	783	749	406	370	195	165
ORG	45	40	105	104	99	92	2	1	79	69	61	54	5	2
LOC	184	182	926	898	696	685	78	75	481	466	269	255	121	113
PERS	46	43	271	238	188	142	31	31	208	199	73	58	56	42
PROD	8	8	43	40	42	35	5	5	15	15	3	3	13	8

Based on the results presented in Table 5, Table 6, Table 7 and Table 8, we consider that the most performing configuration is a monolingual probability table with match correction and the filter 3A for all languages except Finnish and Swedish. For these two languages, it is better to use a multilingual probability table with match correction and filter 3A.[31]

7 Discussion

In this section, we present an analysis with respect to the probability tables used by the EL systems as well discussion regarding the obtained results.

7.1 Probability Tables

This analysis is based on the gold standard data, specifically on the mentions that are linked to a Wikidata entry, i.e. no NILs. The goal is to improve the understanding of the results and the limitations of the proposed methods.

We start the analysis by introducing in Table 9 the number of mentions in the explored corpora that exists

[31] Although, for Finnish and Swedish, the filter 3B seems to provide better results than 3A, in Section 7, we present an analysis of why filter 3A should be considered first.

in each language and that are found in their respective Wikipedia KBs. As it can be observed in Table 9, for all the test splits, except for Swedish, at least 90% (91% − 96%) of the mentions are associated with an entry in the KBs. In comparison, for the Swedish test dataset, only 84% of mentions have a corresponding entry in the KBs. This means that not all the mentions, in a specific language, have a corresponding article in Wikipedia in the language of analysis. For instance, the entity "Porte de Namur" contains a corresponding entry in the Wikidata but not in the Finnish, German, or Swedish Wikipedia KBs. The consequence of this aspect is that by default, monolingual probability tables $p(e|m)$ will not contain all entries necessary to link every mention.

The information presented in Table 9 is as well of relevance because unlike recent works, such as [24,34], we analyze all the mentions even if they do not exist in a KBs. In other words, our EL system is unaware of entities without a corresponding entry in the KBs. This aspect makes the EL task harder to perform, but more realistic, as in many cases, such as the CLEF HIPE 2020 Challenge, it is impossible to know beforehand the entities that will occur. Systems that analyze only mentions found in KBs tend to get better results. Nonetheless, the reason is that these systems reduce the pool of mentions to link and know a priori that these mentions will have a correct match in the KBs.

We present in Table 10, the number of mentions that match their surface form, either exactly or after applying a correction, with an entry in our probability tables $p(e|m)$ (described in Section 3.2). As it can be seen in Table 10, matching entities without applying match corrections (c.f. Subsection 7.3) is quite challenging. For some languages, such as Finnish and Swedish, less than 50% of the mentions match exactly with an entry in the probability tables $p(e|m)$. This shows, that for these languages there is a great variability and complexity on mentions' surface forms, either due to aspects such as inflection and agglutination, or OCR errors found in historical documents. This phenomenon becomes significant on mentions of type person over the Finnish NewsEye dataset, where only 3% of the mentions can be matched in the probability tables.

We can notice in Table 10 that applying a match correction approach (c.f. Subsection 7.3) increases the number of entities that can be found in the probability tables. For instance, in NewsEye Finnish, the word "Berliiniin" (To Berlin) was spelled incorrectly as "Berliniin", however, the match correction module found the correct entry in the KB, "Berliini" (Berlin). In some languages, like Finnish and Swedish, the matching increment is around 60%. Furthermore, we can increase the match of mentions of type person on the Finnish NewsEye dataset from 3% to 25%.

Finally, it is important to highlight that Table 10 allows us determining the maximum number of mentions that can be linked in a dataset if the disambiguation module would be perfect.

7.2 Multilingualism

As presented in Table 10, the use of multilingual probability tables increased the number of mentions that match with an entry in the KBs. However, in multiple cases, the number of new mentions matched is relatively low, with few exceptions within the testing splits for CLEF HIPE 2020 English and, NewsEye French and Swedish. Furthermore, the increment of mention matches is relatively small in comparison to the number of entries added by merging the probability tables in different languages.

The increment on the matches contrasts with the reduction of the performance of the EL systems, in some cases, as shown in Table 5 and Table 6. Based on manual analysis, we have determined three of the causes of this discrepancy.

First, the merge of the probability tables increases the number of possible candidates for each mention. Which, in consequence, requires a more robust EL method, that can deal with the great number of candidates and their possible ambiguity.

Second, the fact that a mention and an entry match, at testing time, according to their surface name, does not ensure the location of a correct link. For instance, certain mentions, such as acronyms, can have different meanings in different languages. Therefore, the EL system might choose the incorrect entry, as it happened with the acronym "UE" that matches "Union Européenne" (European Union) in the French probability table but "University of the East" in the English one.

Third, and due to the nature of historical documents, OCR mistakes along with multilingual probability tables, can increase the ambiguity of entries for a determined mention. For instance, in CLEF HIPE 2020 English, the word France was detected by the OCR as "Fiance"[32]. This caused the EL system using a monolingual probability table to propose a NIL. However, the EL system using a multilingual probability table proposed as candidates "Georges P.Putnam" (Q5543134) and "Engagement" (Q157512).

[32] HIPE-data-v1.3-test-en.tsv#L3070

7.3 Match Correction

The use of match correction has proved to improve the performance of our EL systems as presented in Table 5 and Table 6.

The main reason is that it increases the coverage of the mentions in the probability tables as seen in Table 10. In other words, aspects such as lexical variations, e.g. affixes and inflections, can be measured in order to find the best matching entry. Furthermore, mentions with OCR errors can be more easily linked with their respective entry in the KBs.

However, it must be kept in mind, that the application of a Match Correction can have as well negative side effects. Similar to multilingual probability tables, Match Correction increases the number of entries to disambiguate. In consequence, some mentions might be matched to an incorrect entry.

7.3.1 Filtering

There are three reasons why the post-processing filters improved, in most cases, the performance of the EL systems.

First, the filter fixes redirection pages and removes disambiguation pages. Although both issues are infrequent, their fix can make a difference whether the actual best entry is positioned at the top or not.

Second, adding a NIL before the bottom candidates is a good technique to find mentions that do not have an entry in Wikipedia. However, its effect might not be visible unless we consider more than one candidate during the evaluation. Specifically, the effect of NIL can be seen in Table 7 and Table 8, where we can notice that for some languages, such as CLEF HIPE 2020 English, applying any of the filters resulted in the same score. The only common aspect between all the filters was the addition of a NIL before the bottom candidates. And, it was the addition of the NIL, in most cases, the aspect that improved the performance when evaluating F-score@5.

The results in Table 7 and Table 8, show us as well that the base EL systems have a preference to link most mentions to an entry, rather than proposing a NIL.

Third, placing at the bottom candidates that do not match the mention type according to DBpedia is a good method to improve the performance of the EL system. This can be seen in the fact that positioning candidates not found in DBpedia at the bottom worked better than setting them in the middle.

Apart from the previous aspects, there are some particularities regarding the configuration of the filters that improved the performance of the EL systems.

With respect to the edit distance metric, we have observed that RapidFuzz Weight Ratio[33] produces in general the best ordering of candidates. The reason might be due to the fact that this edit distance metric uses different heuristics, like reordering alphabetically the tokens or scaling the results based on the length of the strings.

There can be as well some other reasons why certain edit distances worked differently on specific datasets. For example, the Weighted Levenshtein might have worked better in English as it uses weights set to fix OCR errors found in English documents [42]. As well, the implementation used only accepts ASCII characters, which might affect languages with diacritics such as French.

Although in Table 6, we observed that not using edit distance performed better on NewsEye Finnish and Swedish, this outcome is caused by exactly two mentions, one in each language. Specifically, the label and/or alternative labels of the entries proposed by the EL systems caused to sort wrongly the top candidates. For instance, in NewsEye Swedish, the EL systems proposed for the mention "Ural" the entries "Uralfloden" (Q80240, Ural River) and "Uralbergen" (Q35600, Ural Mountains). While both entries do not match quotes the mention's surface form, "Uralfloden"[34] has as an alternative label in Swedish the word "Ural", which produces an exact match. This makes the filter set on the first position "Uralfloden" instead of the correct entry "Uralbergen". In the case, of Finnish, for the mention "Eng= lannin" (England), the edit distances considered closer to the entry "Kungariket England" (Q179876, Kingdom of England) rather than "England" (Q21). Based on the fact that only two mentions were affected by this aspect, we consider that on real applications it should always use an edit distance metric to reorder the candidates. And, that this issue is not representative enough to consider that not using an edit distance is a path to follow for these two languages.

Regarding the filtration of entries by date, it is clear that always should be done for mentions of type person. The reason is that most of the Wikipedia entries related to people contain a year of birth. And the gold standard annotators will use as well the year of birth to select the best entry in Wikidata for a person mention.

For the other types of mentions, i.e., location, organization, and product, the performance of the filter by date, seems to depend mostly on the dataset and how well the annotation was done or could be done.

For instance, we noticed that some locations were affected by the date filter due to errors in the gold

[33] https://github.com/maxbachmann/rapidfuzz

[34] https://www.wikidata.org/wiki/Q80240

Table 10: Amount of mentions that match their surface form with an entry existing in the probability tables $p(e|m)$.

(a) CLEF HIPE 2020 dataset.

Splits	English					French					German				
	Mentions	No cor. mono	No cor. multi	Match mono	Correction multi	Mentions	No cor. mono	No cor. multi	Match mono	Correction multi	Mentions	No cor. mono	No cor. multi	Match mono	Correction multi
Train															
Total	–	–	–	–	–	5406	2904	3038	4348	4498	3209	1576	1632	2278	2359
ORG	–	–	–	–	–	554	252	266	465	473	247	82	83	172	173
LOC	–	–	–	–	–	3333	2221	2263	3022	3063	2009	1312	1338	1738	1767
PERS	–	–	–	–	–	1343	355	429	734	829	849	161	187	317	363
PROD	–	–	–	–	–	149	73	77	112	118	77	23	26	51	56
Dev															
Total	549	313	317	418	420	1450	795	840	1135	1167	1157	613	636	845	871
ORG	93	44	45	72	73	120	46	47	97	98	112	19	19	72	72
LOC	332	210	211	272	272	851	595	617	759	767	711	494	504	613	625
PERS	107	50	52	61	62	434	140	158	242	265	294	87	98	131	143
PROD	16	9	9	12	12	37	13	17	30	30	35	13	15	29	31
Test															
Total	283	146	147	202	204	1345	772	796	1118	1137	1025	535	548	735	768
ORG	45	20	20	28	28	105	56	58	90	90	99	37	37	72	73
LOC	184	118	119	149	150	926	610	625	829	842	696	454	460	583	598
PERS	46	8	8	22	23	271	84	90	161	167	188	31	38	57	72
PROD	8	0	0	3	3	43	22	23	38	38	42	13	13	23	25

(b) NewsEye dataset.

Splits	Finnish					French					German					Swedish				
	Mentions	No cor. mono	No cor. multi	Match mono	Cor. multi	Mentions	No cor. mono	No cor. multi	Match mono	Cor. multi	Mentions	No cor. mono	No cor. multi	Match mono	Cor. multi	Mentions	No cor. mono	No cor. multi	Match mono	Cor. multi
Train																				
Total	1281	435	450	843	898	3303	1559	1602	2717	2779	695	371	372	501	505	1447	558	604	999	1107
ORG	92	13	13	54	55	352	106	110	308	31 1	165	21	22	60	60	72	5	5	29	45
LOC	837	398	399	673	691	1786	1174	1201	1662	1681	454	321	321	383	386	834	485	516	722	756
PERS	270	21	35	78	110	1057	244	254	660	697	73	28	28	55	56	395	48	63	153	183
PROD	82	5	5	38	42	108	35	37	87	90	3	1	1	3	3	146	20	20	95	123
Dev																				
Total	133	42	44	84	86	510	295	300	435	443	409	250	253	334	337	199	97	107	158	175
ORG	13	3	3	7	7	57	22	22	51	52	88	46	46	69	69	7	1	1	4	5
LOC	80	37	38	62	62	283	209	211	262	263	223	175	175	209	210	132	81	81	120	123
PERS	31	2	2	13	13	163	59	62	116	122	96	27	30	54	56	49	14	24	24	37
PROD	9	0	1	2	4	7	5	5	6	6	2	2	2	2	2	11	1	1	10	10
Test																				
Total	116	48	49	74	78	783	430	443	656	667	406	220	220	334	339	195	92	100	148	156
ORG	2	0	0	1	1	79	33	36	65	69	61	14	14	49	49	5	1	1	2	2
LOC	78	47	47	65	65	481	304	310	442	444	269	184	184	230	231	121	75	81	108	113
PERS	31	1	2	7	8	208	87	91	135	140	73	22	22	52	56	56	14	16	27	29
PROD	5	0	0	1	4	15	6	6	14	14	3	0	0	3	3	13	2	2	11	12

standard annotation. In Figure 5, we presented the case of the mention "Great Britain" in a press article of 1868.[35] The gold standard annotation indicated that the correct entry is Q145, i.e. United Kingdom (of Great Britain and Northern Ireland). However, because the article was published in 1868, the correct entry should have been Q174193, i.e. United Kingdom of Great Britain and Ireland, which refers to the country that existed before the 1921 Anglo-Irish Treaty. The filter managed to propose the actual correct entry in second place, while removed the entry that matched the gold standard one.

Some other annotation errors are due to the ambiguity of the entry in Wikidata or the impossibility of finding a better candidate. For example, in the French CLEF HIPE 2020, the mention "Val-de-Travers" in a 1798 document is associated in the gold standard to Q70526[36]. Nevertheless, despite the entry has for label "Val-de-Travers" it makes reference to a municipality created in 2009 (field "inception"). Thus, the filter removes it from the candidates. Nonetheless, in Wikipedia, it does not seem to exist a better candidate to annotate the entry. Some of the other entries, such as "Val-de-Travers District" or "Region of Val-de-Travers" make reference to relative modern locations too.

From a detailed analysis, we observed that most of the mention types were benefited from the application of filters. The exception was those belonging to organizations, in which the filter decreased the number of mentions with a correct entry positioned in the first place. Nonetheless, when we evaluate the performance using F-score@5, this discrepancy is no longer observable. This means, that the correct entry for organizations tends to be misplaced. The most probable reason is the small number of associated DBpedia types related to organizations as described in Table 1. As well, it can be related to a small coverage of organizations in DBpedia and DBpedia chapters.

[35] It should be noted that the context surrounding the mention, indicates that "Great Britain" is referring to a country and not the island.

[36] https://www.wikidata.org/wiki/Q70526

8 Conclusion and Future Work

Historical documents are an essential source regarding cultural and historical heritage of countries, regions, and languages. With their digitization, the accessibility to these documents has increased considerably, together with the need for information that can enrich these documents.

To enrich historical documents, digital humanities researchers have approached the natural language processing (NLP) community in order to have access to tools such as named entity recognition and entity linking. And, although the use of NLP tools has expanded to multiple domains and types of documents, their use in historical corpora has been limited. Aspects such as optical character recognition (OCR) errors and spelling variations, which make NLP tasks harder to perform, have limited the number of tools available for historical documents.

In order to fill this gap, we presented MELHISSA, a Multilingual Entity Linking architecture for HIstorical preSS Articles. The main objective of this tool is to link mentions, such as names of people, organizations, and products, to entries in knowledge bases, such as Wikidata. Specifically, we created an end-to-end neural entity linking system, that manages multiple languages and has been designed to surpass common errors found in historical documents.

The presented system was tested over two historical datasets, NewsEye and CLEF HIPE 2020, comprising five European languages: English, Finnish, French, German, and Swedish. We explored different configurations, such as the use of edit distances, multilingual probability tables, and post-processing filters, in order to create a performing entity linking tool.

The obtained outcomes demonstrated that MELHISSA is a competitive tool that is able to get an F-score@1 of up to 0.655 and an F-score@5 of up to 0.752. We have observed that the use of multilingual probability tables can be useful in languages such as Finnish and Swedish, while the use of a matching correction module can improve the pairing of mentions and entries in a knowledge base. Furthermore, the application of a post-processing filter can improve in general the entity linking performances in all languages.

In the future, we would like to apply MELHISSA to other languages and to contemporary documents. One extension could explore diachronic embeddings to see if these can improve the matching of entities that their spelling has evolved through time. Finally, a comparative study of these embeddings on historical and contemporary documents could prove the effectiveness of MELHISSA steps (match correction, filtering) on larger periods.

Conflict of interest

The authors declare that they have no conflict of interest.

Author contributions

Elvys Linhares Pontes: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing. **Luis Adrián Cabrera-Diego** Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing. **Jose G. Moreno:** Conceptualization, Methodology, Project administration, Validation, Supervision, Writing - review & editing. **Emanuela Boros:** Supervision, Writing - review & editing. **Ahmed Hamdi** Formal analysis, Investigation, Data Curation, Visualization, Writing - review & editing. **Antoine Doucet:** Funding acquisition, Conceptualization, Methodology, Project administration, Validation, Supervision, Writing - review & editing. **Nicolas Sidere:** Supervision, Validation, Writing - review & editing.

Availability of code, data and material

Our code is freely available at `https://github.com/NewsEye/Named-Entity-Linking/tree/master/multilingual_entity_linking` for the EL method and at `https://github.com/EMBEDDIA/NEL_Filter` for the filter module. The CLEF HIPE 2020 dataset is available at `https://impresso.github.io/CLEF-HIPE-2020/` and the NewsEye dataset at `https://zenodo.org/record/4573313#.YH79nnUzY5k`.

Acknowledgements This work has been supported by the European Union's Horizon 2020 research and innovation programme under grants 770299 (NewsEye) and 825153 (EMBEDDIA).

References

1. Gazette of the United-States. (New York, New York, U.S.A). In: Chronicling America: Historic American Newspapers. Library of Congress (02-Jan-1790). URL `https://chroniclingamerica.loc.gov/lccn/sn83030483/1790-01-02/ed-1/seq-4/`. Accessed on April 2021

2. Gazette of the United-States. (New York, New York, U.S.A). In: Chronicling America: Historic American Newspapers. Library of Congress (03-Mar-1790). URL https://chroniclingamerica.loc.gov/lccn/sn83030483/1790-03-03/ed-1/seq-4/. Accessed on April 2021
3. Vossische Zeitung. (Berlin , Germany). Staatsbibliothek zu Berlin (11-Feb-1857). URL https://dfg-viewer.de/show/?set%5Bmets%5D=https://content.staatsbibliothek-berlin.de/zefys/SNP27112366-18570211-0-0-0-0.xml. Accessed on April 2021
4. Chariton Courier. (Keytesville, Chariton County, Missouri, U.S.A). In: Chronicling America: Historic American Newspapers. Library of Congress (13-Feb-1890). URL https://chroniclingamerica.loc.gov/lccn/sn88068010/1890-02-13/ed-1/seq-3/. Accessed on April 2021
5. Le Libérateur du Sud-Ouest : organe régional du Parti populaire français. (Bordeaux , France). Bibliothèque nationale de France (3-Dec-1936). URL https://gallica.bnf.fr/ark:/12148/bpt6k55631820. Accessed on April 2021
6. Les Affiches de Paris (Paris , France). Bibliothèque nationale de France (31-Dec-1750). URL https://gallica.bnf.fr/ark:/12148/bpt6k10531388. Accessed on April 2021
7. Abramitzky, R., Mill, R., Pérez, S.: Linking individuals across historical sources: a fully automated approach. Historical Methods: A Journal of Quantitative and Interdisciplinary History **53**(2), 94–111 (2020)
8. Agirre, E., Barrena, A., de Lacalle, O.L., Soroa, A., Fernando, S., Stevenson, M.: Matching cultural heritage items to wikipedia. In: Eight International Conference on Language Resources and Evaluation (LREC) (2012)
9. Bair, S., Carlson, S.: Where keywords fail: Using metadata to facilitate digital humanities scholarship. Journal of library Metadata **8**(3), 249–262 (2008)
10. Boroş, E., Hamdi, A., Pontes, E.L., Cabrera-Diego, L.A., Moreno, J.G., Sidere, N., Doucet, A.: Alleviating digitization errors in named entity recognition for historical documents. In: Proceedings of the 24th Conference on Computational Natural Language Learning, pp. 431–441 (2020)
11. Boros, E., Linhares Pontes, E., Cabrera-Diego, L.A., Hamdi, A., Moreno, J.G., Sidère, N., Doucet, A.: Robust Named Entity Recognition and Linking on Historical Multilingual Documents. In: L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
12. Boroş, E., Romero, V., Maarand, M., Zenklová, K., Křečková, J., Vidal, E., Stutzmann, D., Kermorvant, C.: A comparison of sequential and combined approaches for named entity recognition in a corpus of handwritten medieval charters. In: 2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 79–84. IEEE (2020)
13. Brando, C., Frontini, F., Ganascia, J.G.: Disambiguation of Named Entities in Cultural Heritage Texts Using Linked Data Sets. In: T. Morzy, P. Valduriez, L. Bellatreche (eds.) First International Workshop on Semantic Web for Cultural Heritage, SW4CH 2015, *Communications in Computer and Information Science*, vol. 539, pp. 505–514. Springer, Poitiers, France (2015). DOI 10.1007/978-3-319-23201-0_51. URL https://hal.archives-ouvertes.fr/hal-01203784
14. Brando, C., Frontini, F., Ganascia, J.G.: REDEN: Named Entity Linking in Digital Literary Editions Using Linked Data Sets. Complex Systems Informatics and Modeling Quarterly **2016**(7), 60 – 80 (2016). DOI 10.7250/csimq.2016-7.04. URL https://hal.sorbonne-universite.fr/hal-01396037
15. Broscheit, S.: Investigating entity knowledge in bert with simple neural end-to-end entity linking. arXiv preprint arXiv:2003.05473 (2020)
16. Chen, S., Wang, J., Jiang, F., Lin, C.Y.: Improving entity linking by modeling latent entity type information. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 7529–7537 (2020)
17. Chiron, G., Doucet, A., Coustaty, M., Visani, M., Moreux, J.P.: Impact of ocr errors on the use of digital libraries: towards a better access to information. In: Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries, pp. 249–252. IEEE Press (2017)
18. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 708–716. Association for Computational Linguistics, Prague, Czech Republic (2007). URL https://www.aclweb.org/anthology/D07-1074
19. De Wilde, M.: Improving retrieval of historical content with entity linking. In: T. Morzy, P. Valduriez, L. Bellatreche (eds.) New Trends in Databases and Information Systems (ADBIS 2015), pp. 498–504. Springer International Publishing (2015). DOI 10.1007/978-3-319-23201-0_50
20. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
21. Ehrmann, Romanello, Clematide, Flückiger: HIPE - Shared Task Participation Guidelines (2020). DOI 10.5281/zenodo.3677171. URL https://doi.org/10.5281/zenodo.3677171
22. Ehrmann, M., Romanello, M., Bircher, S., Clematide, S.: Introducing the CLEF 2020 HIPE shared task: Named entity recognition and linking on historical newspapers. In: J.M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M.J. Silva, F. Martins (eds.) Proceedings of the 42nd European Conference on IR Research (ECIR 2020), vol. 2, pp. 524–532. Springer International Publishing (2020). DOI 10.1007/978-3-030-45442-5_68
23. Frontini, F., Brando, C., Ganascia, J.G.: Semantic web based named entity linking for digital humanities and heritage texts. In: Proceedings of the First International Workshop Semantic Web for Scientific Heritage at the 12th ESWC 2015 Conference, vol. 1364 (2015)
24. Ganea, O.E., Hofmann, T.: Deep joint entity disambiguation with local neural attention. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2619–2629. Association for Computational Linguistics (2017). DOI 10.18653/v1/D17-1277
25. Gefen, A.: Les enjeux épistémologiques des humanités numériques. Socio (2015). DOI https://doi.org/10.4000/socio.1296
26. Hamdi, A., Boroş, E., Pontes, E.L., Nguyen, T.T.H., Hackl, G., Moreno, J.G., Doucet, A.: A multilingual dataset for named entity recognition, entity linking and stance detection in historical newspapers. In: Proceedings of the 44rd International ACM SIGIR Conference

on Research and Development in Information Retrieval (2021)
27. Hechl, S., Langlais, P.C., Marjanen, J., Oberbichler, S., Pfanzelter, E.: Digital interfaces of historical newspapers: opportunities, restrictions and recommendations. Journal of Data Mining & Digital Humanities (2021)
28. Heino, E., Tamper, M., Mäkelä, E., Leskinen, P., Ikkala, E., Tuominen, J., Koho, M., Hyvönen, E.: Named entity linking in a complex domain: Case second world war history. In: J. Gracia, F. Bond, J.P. McCrae, P. Buitelaar, C. Chiarcos, S. Hellmann (eds.) Language, Data, and Knowledge, pp. 120–133. Springer International Publishing, Galway, Ireland (2017). DOI 10.1007/978-3-319-59888-8_10
29. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997). DOI 10.1162/neco.1997.9.8.1735
30. Hoffart, J., Yosef, M.A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust disambiguation of named entities in text. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 782–792. Association for Computational Linguistics, Edinburgh, Scotland, UK. (2011). URL https://www.aclweb.org/anthology/D11-1072
31. van Hooland, S., De Wilde, M., Verborgh, R., Steiner, T., Van de Walle, R.: Exploring entity recognition and disambiguation for cultural heritage collections. Digital Scholarship in the Humanities **30**(2), 262–279 (2013). DOI 10.1093/llc/fqt067
32. Huet, T., Biega, J., Suchanek, F.M.: Mining history with le monde. In: Proceedings of the 2013 Workshop on Automated Knowledge Base Construction, AKBC '13, p. 49–54. Association for Computing Machinery, New York, NY, USA (2013). DOI 10.1145/2509558.2509567
33. Klie, J.C., de Castilho, R.E., Gurevych, I.: From zero to hero: Human-in-the-loop entity linking in low resource domains. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 6982–6993 (2020)
34. Kolitsas, N., Ganea, O.E., Hofmann, T.: End-to-end neural entity linking. In: Proceedings of the 22nd Conference on Computational Natural Language Learning, pp. 519–529. Association for Computational Linguistics (2018). DOI 10.18653/v1/K18-1050
35. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., Kleef, P.v., Auer, S., Bizer, C.: DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web Journal **6**(2), 167–195 (2015). DOI 10.3233/SW-140134
36. Linhares Pontes, E., Hamdi, A., Sidere, N., Doucet, A.: Impact of OCR quality on named entity linking. In: Digital Libraries at the Crossroads of Digital Information for the Future - 21st International Conference on Asia-Pacific Digital Libraries, ICADL 2019, Kuala Lumpur, Malaysia, November 4-7, 2019, Proceedings, pp. 102–115 (2019). DOI 10.1007/978-3-030-34058-2_11
37. Linhares Pontes, E., Moreno, J.G., Doucet, A.: Linking named entities across languages using multilingual word embeddings. In: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, JCDL '20, p. 329–332. Association for Computing Machinery, New York, NY, USA (2020). DOI 10.1145/3383583.3398597. URL https://doi.org/10.1145/3383583.3398597
38. Moreno, J.G., Besançon, R., Beaumont, R., D'hondt, E., Ligozat, A.L., Rosset, S., Tannier, X., Grau, B.: Combining word and entity embeddings for entity linking. In: European Semantic Web Conference, pp. 337–352. Springer (2017)
39. Mosallam, Y., Abi-Haidar, A., Ganascia, J.G.: Unsupervised named entity recognition and disambiguation: An application to old french journals. In: P. Perner (ed.) Advances in Data Mining. Applications and Theoretical Aspects, pp. 12–23. Springer International Publishing, St. Petersburg, Russia (2014). DOI 10.1007/978-3-319-08976-8_2
40. Munnelly, G., Lawless, S.: Investigating entity linking in early english legal documents. In: Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries, JCDL'18, p. 59–68. Association for Computing Machinery, New York, NY, USA (2018). DOI 10.1145/3197026.3197055
41. Munnelly, G., Pandit, H.J., Lawless, S.: Exploring linked data for the automatic enrichment of historical archives. In: European Semantic Web Conference, pp. 423–433. Springer (2018). DOI 10.1007/978-3-319-98192-5_57
42. Nguyen, N.K., Boros, E., Lejeune, G., Doucet, A.: Impact analysis of document digitization on event extraction. In: 4th Workshop on Natural Language for Artificial Intelligence (NL4AI 2020) co-located with the 19th International Conference of the Italian Association for Artificial Intelligence (AI* IA 2020), vol. 2735, pp. 17–28 (2020)
43. Nguyen, T.T.H., Jatowt, A., Coustaty, M., Nguyen, N.V., Doucet, A.: Deep statistical analysis of ocr errors for effective post-ocr processing. In: Proceedings of the 18th Joint Conference on Digital Libraries, JCDL '19, p. 29–38. IEEE Press (2019). DOI 10.1109/JCDL.2019.00015. URL https://doi.org/10.1109/JCDL.2019.00015
44. Oberbichler, S., Pfanzelter, E., Marjanen, J., Hechl, S.: Doing historical research with digital newspapers – perspectives of dh scholars. EuropeanaTech Insight, 16: Newspapers (2020). URL https://pro.europeana.eu/page/issue-11-generous-interfaces
45. Onoe, Y., Durrett, G.: Fine-grained entity typing for domain independent entity linking. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 8576–8583 (2020)
46. Pellissier Tanon, T., Weikum, G., Suchanek, F.: YAGO 4: A reason-able knowledge base. In: A. Harth, S. Kirrane, A.C. Ngonga Ngomo, H. Paulheim, A. Rula, A.L. Gentile, P. Haase, M. Cochez (eds.) Proceedings of the 17th International Conference, ESWC 2020, The Semantic Web, pp. 583–596. Springer International Publishing (2020). DOI 10.1007/978-3-030-49461-2_34
47. Pontes, E.L., Cabrera-Diego, L.A., Moreno, J.G., Boros, E., Hamdi, A., Sidère, N., Coustaty, M., Doucet, A.: Entity linking for historical documents: Challenges and solutions. In: E. Ishita, N.L.S. Pang, L. Zhou (eds.) Digital Libraries at Times of Massive Societal Transition, pp. 215–231. Springer International Publishing, Cham (2020)
48. Rijhwani, S., Xie, J., Neubig, G., Carbonell, J.: Zero-shot neural transfer for cross-lingual entity linking. In: Thirty-Third AAAI Conference on Artificial Intelligence (AAAI). Honolulu, Hawaii (2019). DOI 10.1609/aaai.v33i01.33016924
49. Ruiz, P., Poibeau, T.: Mapping the Bentham Corpus: Concept-based Navigation. Journal of Data Mining and Digital Humanities. **Special Issue: Digital Humanities between knowledge and know-how (Atelier Digit_Hum)** (2019). URL https://hal.archives-ouvertes.fr/hal-01915730

50. Sang, E.F., De Meulder, F.: Introduction to the conll-2003 shared task: Language-independent named entity recognition. arXiv preprint cs/0306050 (2003)
51. Shen, W., Wang, J., Han, J.: Entity linking with a knowledge base: Issues, techniques, and solutions. IEEE Transactions on Knowledge and Data Engineering **27**(2), 443–460 (2015). DOI 10.1109/TKDE.2014.2327028
52. Smith, D.A., Crane, G.: Disambiguating geographic names in a historical digital library. In: Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries, ECDL '01, p. 127–136. Springer-Verlag, Darmstadt, Germany (2001). DOI 10.1007/3-540-44796-2_12
53. Wevers, M., Koolen, M.: Digital begriffsgeschichte: Tracing semantic change using word embeddings. Historical Methods: A Journal of Quantitative and Interdisciplinary History **53**(4), 226–243 (2020)
54. Zhou, S., Rijhwani, S., Neubig, G.: Towards zero-resource cross-lingual entity linking. In: Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019), pp. 243–252. ACL, China (2019). DOI 10.18653/v1/D19-6127
55. Zhou, S., Rijhwani, S., Wieting, J., Carbonell, J., Neubig, G.: Improving candidate generation for low-resource cross-lingual entity linking. Transactions of the Association for Computational Linguistics **8**, 109–124 (2020)

E Named entity recognition architecture combining contextual and global features

Named entity recognition architecture combining contextual and global features

No Author Given

No Institute Given

Abstract. Named entity recognition (NER) is an information extraction technique that aims to locate and classify named entities within a document into predefined categories. Entities include organizations, locations, currencies, quantities, and many more. Correctly identifying these phrases is a difficult task because they have multiple forms and they are context dependent. While the context can strongly be represented by contextual features, the global relations are often misrepresented by those models. In this paper, we propose the combination of contextual features from XLNet and global features from Graph Convolution Network (GCN) to enhance NER performance. Experiments over a widely-used dataset, CoNLL 2003, show the benefits of our strategy, outperforming the state of the art.

1 Introduction

Named entity recognition (NER) or entity extraction is an information extraction technique that aims to locate named entities in text and classify them into predefined categories (organizations, locations, quantities, etc). Correctly identifying entities plays an important role in natural language understanding and numerous downstream applications such as relation extraction, entity linking, question answering, or machine translation.

Traditionally, NER was a challenging task, requiring huge amounts of knowledge in the form of feature engineering and lexicons as well as appropriate rules to improve the performance. High performing statistical approaches have been used to predict the entities, notably Markov models, Conditional Random Fields (CRFs), and Support Vector Machines (SVMs). Deep learning works have brought the use of neural networks closer to NER with competitive achievements thanks to the consideration of contextual information, from a simple fixed size window feed-forward network to the state-of-the-art (SOTA) Recurrent Neural Networks (RNNs). The key improvements include using a bi-directional Long Short-Term Memory (LSTM) in place of a feed-forward network and concatenating morphological information to the input vectors, not to mention the recent emergence of Transformer to achieve the SOTA solutions for several sequence tasks including NER.

A crucial component that contributes to the success of NER progress is how meaningful information can be captured from original data via the word embeddings, which can be divided into 2 major types: global features and contextual features (in the scope of this paper, the word "features" and "embeddings" are interchangeable terms).

- **Global features** [33] capture latent syntactic and semantic similarities. They are first constructed from a global vocabulary (or dictionary) of unique words in the documents. Then, similar representations are learnt based on how frequently the words appear close to each other. The problem of such features is that the words' meaning in varied contexts is often ignored. That means, given a word, its embedding always stays the same in whichever sentence it occurs. Due to this characteristic, we can also define global features as "*static*". Some examples are word2vec, GloVe, FastText, to mention a few.

– **Contextual features** [7, ?] capture word semantics in context to address the polysemous and context-dependent nature of words. By passing the entire sentence to the pretrained model, we assign each word a representation based on its context, then capture the uses of words across different contexts. Thus, given a word, the contextual features are dynamically generated instead of being static as the global one. Some examples are ELMo, BERT, XLNet, to mention a few.

In term of global features, there exist several tokens that are always parts of an entity. As an example, in the CoNLL 2003 dataset, the token "Cup" appears 95 times and all cases correspond to entity mentions although the word "Cup" is not an entity on its own. More obvious cases are the names of countries include U.S. (377 mentions), Germany (143 mentions), Australia (136 mentions), Britain (133 mentions), England (127 mentions), France (127 mentions), to mention a few. However, it is not true to all tokens in an entity. The token may or may not be part of an entity (i.e "The White House" vs. "the white windows" and "Jobs said" vs. "Jobs are hard to find") and may belong to different entity types depending on the context of the sentence (i.e "Washington" can be classified as a person or a location).

Meanwhile, the contextual features or contextual information are based on neighboring tokens, as well as the token itself. They aim to represent word semantics in context aiming to solve the problem of using global features, so as to improve the prediction performance (i.e "Jobs" in "Jobs said" and "Jobs are hard to find" will have different representations). Numerous recent studies on sequence labelling tasks in general and NER, in particular, take advantage of contextual information.

Nowadays, NER is still a demanding task because the entities have multiple forms and are context dependent. Most recent research investigates contextual features and often misrepresents the global relations. In this paper, we present a joint architecture to enhance the performance of NER. Extensive experiments on the CoNLL 2003 dataset suggest that our strategy surpasses the systems with standalone feature representation (either global or contextual one). The main contributions of this paper can be summarized as follows:

– We introduce a new architecture that combines the contextual features from XLNet and the global features from GCN to enhance NER performance.
– We demonstrate that our model outperforms the systems using only contextual or global features alone on NER.
– We report new SOTA results on the CoNLL 2003 dataset.

This paper is organised as follows: Section 2 presents the related work in NER. It leads to the full description of our approach in Section 3, with the corresponding experimental framework presented in Section 4. The corresponding results are detailed in Section 5, before we conclude and present future leads in Section 6.

2 Related work

2.1 Named entity recognition

The term "Named Entity" (NE) has first appeared in the sixth Message Understanding Conference (MUC-6) [10] to define the recognition of the information units such as names of organizations, people, geographic locations, currencies, time, and percentage expressions. Regarding the surveys on diverse techniques applied to NER tasks [32, 43, 20], we can broadly divide the NER approaches into four categories: Rule-based, unsupervised learning, feature-based supervised learning, and deep learning based approaches.

Rule-based NER tagging Rule-based NER is the most traditional technique that does not require annotated data as it relies on manually-crafted rules (e.g. LTG [30], NetOwl [16]). These rules are designed by experts based on the syntactic and lexical patterns, linguistics and domain knowledge. Despite good performance when the lexicon is exhaustive, such systems often achieve high precision and low recall due to the limitation on domain-specific rules and incomplete dictionaries.

Unsupervised learning based NER tagging Another approach that also needs no annotated data is unsupervised learning, typically NE clustering [5]. The key idea is to extract NEs from the clustered groups based on context similarity. The lexical resources, lexical patterns, and statistics are computed on a large corpus and then applied to infer mentions of NEs. Several works proposed the unsupervised systems for NE ambiguity to extract named entities in diverse domains [9, 31], especially in biomedical text.

Supervised based NER tagging In supervised NER, given annotated data, features are carefully designed so that the machine learning model can learn to recognize similar patterns from unseen data. Feature engineering plays a key role in improving the performance of the supervised systems. There are numerous ways to represent features, including global information such as word-level features [23], lookup features [11], document and corpus features [12], and many more. Several statistical methods have been used to predict the entities, notably Markov models, CRFs, and SVMs. Among these algorithms above, CRF-based NER has been widely applied to identify entities from texts in various domains, including biomedical text [25], tweets [36] and chemical text [37]. However, these mentioned approaches depend heavily on hand-crafted features and domain-specific resources, which results in the difficulty to adapt to new tasks or to transfer to new domains.

Deep Learning based NER tagging In recent years, the revolution of deep learning has brought the use of neural networks closer to NER tasks with significant achievements. Unlike statistical methods, neural networks offer the non-linear transformation so that the deep learning models can learn complex features and discover useful representations as well as underlying factors. Neural architectures for NER often make use of the combination of either RNNs and CRFs [4] or Convolution Neural Networks (CNNs) and CRFs to extract information automatically from the inputs and detect NER labels. With further researches on contextual features, RNNs with Long Short-Term Memory units (LSTMs) and CRFs have been proposed [17] to improve the performance on identifying NER tags significantly. Moreover, the conjunction of bidirectional LSTMs, CNNs, and CRFs [29] is also introduced to exploit both word- and character-level representations automatically. Meanwhile, the combination of neural language models (i.e ELMo, BERT), LSTMs, and CRFs [24] is applied to extract knowledge from raw texts and empower the sequence labeling tasks as NER.

2.2 Embeddings

A key factor that contributes to the success of NER tasks is how we capture meaningful information from original data via the word representations, especially global features and contextual features.

Global features Global features are context-free word representations that can capture meaningful semantic and syntactic information. It can be represented at different levels such as word-level features (i.e morphology, POS tag) [23], lookup features [11], document and corpus features (i.e local syntax, multiple occurrences) [12] . The impact of global features has been demonstrated in several ways, i.e feeding the co-occurrence of each token [3] into a maximum entropy classifier, encoding them into hidden states of bidirectional RNN, or re-ranking NER [44] to leverage global sentence patterns. Recently, the global sentence-level representation [49]

has been proposed to capture global features more precisely and it outperforms on various sequence labeling tasks. Furthermore, the Graph Neural Network [47] is getting attention to not only have rich relational structure but also preserve global structure information of a graph in graph embeddings.

Contextual features Different from previous word embeddings, the contextual features are context-aware word representations that can capture word semantics under diverse linguistic contexts. That is, a word can be represented differently and dynamically under particular circumstance. The contextual embeddings are often pretrained on large-scale unlabelled corpora and can be divided into 2 types: unsupervised approaches [18, 19] and supervised approaches [6, 39]. While global embeddings ignore the meaning of the word, contextual embeddings succeed in exploring and exploiting the polysemous and context-dependent nature of words, thereby moving beyond global word embeddings and contributing significant improvements in NER. Unlike the contextual embeddings which can disambiguate word meanings and achieve ground-breaking performance, global features are still underrepresented.

3 Methodology

In this section, we explain how we extract global as well as contextual features and how to combine them. For global features, we take advantage of graph representations using GCN [38, 2], which is used to better capture the correlation between NEs and the global semantic information in text and to avoid the loss of detailed information. In addition, we use XLNet [45], a pretrained language model to capture contextual features. Employing Transformer-XL as the backbone model, XLNet exhibits excellent performance for language tasks by learning from bi-directional context and becomes a competitive pretrained language model for NER. The details are explained in the following subsections.

3.1 GCN as Global Embeddings

Graph Convolutional Network (GCN) aims to learn a function of signals/features on a graph $G = (V, E)$ with V as Vertices and E as Edges. Given N as number of nodes, D as number of input features, and F as the number of output features per node, GCN takes 2 inputs:

- An $N \times D$ feature matrix X as feature description.
- An adjacency matrix A as representative description of the graph structure.

and returns an $N \times F$ feature matrix as the output Z [15, 8].

Every neural network layer can then be written in the form of a non-linear function:

$$H^{(l+1)} = f(H^{(l)}, A) \tag{1}$$

where $H^{(0)} = X$, $H^{(L)} = Z$, L being the number of layers. The specific models then differ only in how we choose and parameterize $f(.,.)$.

In our specific task, we capture the global features by feeding feature matrix X and adjacent matrix A into a graph using two-layer spectral convolutions introduced in Graph Convolutional Network (GCN) [15]. Raw texts are first transformed into word embeddings using GloVe, an unsupervised learning algorithm to obtain vector representations for words. Then, Universal Dependencies are employed so that the input embeddings are converted into graph embeddings where words become nodes and dependencies become edges. After that, two-layer GCN is applied to the generated matrix of nodes feature vectors X and the adjacent matrix A to extract meaningful global features.

Mathematically, given a specific graph-based neural network model $f(X, A)$, spectral GCN follows layer-wise propagation rule:

$$H^{(l+1)} = \sigma(\tilde{D}^{\frac{-1}{2}}\tilde{A}\tilde{D}^{\frac{-1}{2}}H^{(l)}W^{(l)}) \tag{2}$$

where A is the adjacency matrix, X is the matrix of node feature vectors (given sequence x), D is the degree matrix, $f(\cdot)$ is the neural network like differentiable function, $\tilde{A} = A + I_N$ is the adjacency matrix of the undirected graph G with added self-connections, I_N is the identity matrix of N nodes, $\tilde{D}_i = \sum_j \tilde{A}_{ij}$, $W^{(l)}$ is the layer-specific trainable weight matrix, $\sigma(\cdot)$ is the activation function, and $H^{(l)} \in \mathbb{R}^{(N\times D)}$ is the matrix of activation in the l^{th} layer (representation of the l^{th} layer), $H^{(0)} = X$.

After calculating the normalized adjacency matrix $\tilde{D}^{\frac{-1}{2}}\tilde{A}\tilde{D}^{\frac{-1}{2}}$ in preprocessing step, the forward model can be expressed as:

$$Z = f(X, A) = softmax(\tilde{A}ReLU(\tilde{A}XW^0)W^1) \tag{3}$$

where $W^{(0)} \in R^{C\times H}$ is the input-to-hidden weight matrix for a hidden layer with H feature maps and $W^{(1)} \in R^{H\times F}$ is the hidden-to-output weight matrix.

$W^{(0)}$ and $W^{(1)}$ are trained using gradient descent. The weights before feeding to Linear layer with Softmax activation function are taken as global features to feed in our combined model. We keep the prediction results of GCN after feeding weights to the last Linear layer to compare the performance and prediction qualities with our proposed architecture's results.

3.2 XLNet as Contextual Embeddings

XLNet is an autoregressive (AR) pretraining method based on a novel generalized permutation language modeling objective. Employing Transformer-XL as the backbone model, XLNet is a break-though that exhibits excellent performance for language tasks involving long context such as NER by learning from bi-directional context and avoiding the disadvantages brought by the MASK method in Autoencoding (AE) language model.

The contextual features are captured from the sequence using permutation language modeling objective and two-stream self-attention architecture, integrating relative positional encoding scheme and the segment recurrence mechanism from Transformer-XL [45]. Given a sequence x of length T, the permutation language modeling objective can be defined as:

$$\max_{\theta} \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}_T}\left[\sum_{t=1}^{T}\log p_\theta\left(x_{z_t} \mid \mathbf{x}_{\mathbf{z}<t}\right)\right] \tag{4}$$

where $\mathcal{Z}_T$ is the set of all possible permutations of the index sequence of length T $[1, 2, ..., T]$, z_t is the t^{th} element of a permutation $\mathbf{z} \in \mathcal{Z}_T$, $\mathbf{z} < t$ is the first $(t-1)^{th}$ elements of a permutation $\mathbf{z} \in \mathcal{Z}_T$, and p_θ is the likelihood. θ is the parameter shared across all factorization orders during training so x_t is able to see all $x_i \neq x_t$ possible elements in the sequence.

We also use two-stream self-attention to remove the ambiguity in target predictions. For each self-attention layer $m = 1, ..., M$, the two streams of representation are updated schematically with a shared set of parameters:

$$\begin{aligned} g_{z_t}^{(m)} &\leftarrow Attention\left(\mathrm{Q} = g_{z_t}^{(m-1)}, \mathrm{KV} = \mathbf{h}_{\mathrm{z}<t}^{(m-1)}; \theta\right) \\ h_{z_t}^{(m)} &\leftarrow Attention\left(\mathrm{Q} = h_{z_t}^{(m-1)}, \mathrm{KV} = \mathbf{h}_{\mathrm{z}\leq t}^{(m-1)}; \theta\right) \end{aligned} \tag{5}$$

where $g_{z_t}^{(m)}$ is the query stream that uses z_t but cannot see x_{z_t}, $h_{z_t}^{(m)}$ is the content stream that uses both z_t and x_{z_t}, and K, Q, V are the key, query, value, respectively.

To avoid slow convergence, the objective is customized to maximize the log-likelihood of the target sub-sequence conditioned on the non-target sub-sequence as in Equation 6.

$$\max_{\theta} \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}_T} \left[\log p_\theta \left(x_{\mathbf{z}_{>c}} \mid \mathbf{x}_{\mathbf{z}_{\leq c}}\right)\right] = \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}_T} \left[\sum_{t=c,the+1}^{|z|} \log p_\theta \left(x_{z_t} \mid \mathbf{x}_{\mathbf{z}<t}\right)\right] \tag{6}$$

where $\mathbf{z}_{>c}$ is the target sub-sequence, $\mathbf{z}_{\leq c}$ is the non-target sub-sequence, and c is the cutting point.

Furthermore, we make use of relative positional encoding scheme and the segment recurrence mechanism from Transformer-XL. While the position encoding ensures the reflection in the positional information of text sequences, the attention mask is applied so the texts are given different attention during the creation of input embedding. Given 2 segments $\mathbf{x} = s_{1:T}$ and $\mathbf{x} = s_{T:2T}$ from a long sequence s, $\mathbf{z}$ and z referring the permutations of [1 ... T] and [T + 1 ... 2T], we process the first segment, and then cache the obtained content representations $\mathbf{h}^{(m)}$ for each layer m. After that, we update the attention for the next segment x with memory, which can be expressed as in Equation 7.

$$h_{z_t}^{(m)} \leftarrow \mathit{Attention}\left(\mathrm{Q} = h_{z_t}^{(m-1)}, \mathrm{KV} = \left[\tilde{\mathbf{h}}^{(m-1)}, \mathbf{h}_{z\leq t}^{(m-1)}\right]; \theta\right) \tag{7}$$

Similar to global features, we capture the weights before feeding to the last Linear layer and use it as contextual embeddings of our combined model. For the purpose of comparison, we also keep the prediction results of XLNet after feeding weights to the last Linear layer.

3.3 Joint Architecture

Given global and contextual features extracted from GCN and XLNet, respectively, we simply concatenate and feed them into a Linear layer. We choose the simplest way to show the most evident impact of the global and contextual features to the NER task. The proposed approach is presented in Fig. 1.

4 Experimental setup

In this section, we present our experimental configuration in details. First, we describe the dataset and the evaluation metrics. Then, we present the XLNet and GCN implementations, as well as the proposed joint model.

4.1 Dataset and metrics

We opted for the CoNLL 2003 dataset [40], one of the widely-adopted benchmark datasets for NER tasks. The English version is collected from the Reuters Corpus[1] with news stories between August 1996 and August 1997. The dataset concentrates on 4 types of named entities: persons (PER), locations (LOC), organizations (ORG), and miscellaneous (MISC). The latter group are named entities that do not belong to the previous three groups. Standard Precision, Recall and F1-measure as well as standard data splits were used in all the presented results.

[1] http://www.reuters.com/researchandstandards/

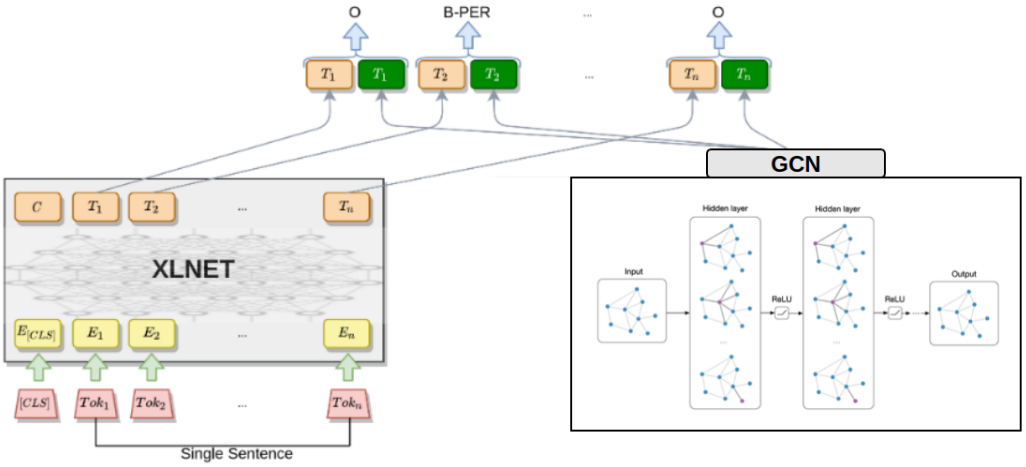


Fig. 1: Visualization of the global architecture of our proposed approach.

4.2 Implementation details

Global embeddings with GCN The sentences are annotated with dependency parser (universal dependencies from Spacy library) to create a graph of relations between the words, where words become nodes, and dependencies become edges. CoNLL 2003 dataset is converted into 124 nodes and 44 edges with the training corpus size of approximately 2 billion words, the vocabulary size of 222,496, and the dependency context vocabulary size of 1,253,524. The graph embeddings are then fed into 2 Graph Convolution layers with a Dropout layer of 0.5 after each layer to avoid overfitting. The global features are then captured before the last Linear layer. We perform batch gradient descent using the whole dataset for every training iteration, which is a feasible option as long as the dataset fits in memory. We also take advantage of TensorFlow for efficient GPU-based implementation of Equation 2 using sparse-dense matrix multiplications.

Contextual embeddings with XLNet We have investigated on diverse embeddings such as FastText[2], Flair[3], Stanza[4] and XLNet[5] pretrained embeddings. Preliminary results suggest that XLNet (XLNet-Base, Cased) outperforms other representations, and is therefore chosen for our final implementation. The word embedding of size 768 with 12 layers were used for XLNet. Each layer consists of 3 sublayers, including XLNet Relative Attention, XLNet Feed Forward, and Dropout layer. The XLNet Relative Attention is a two-stream self-attention mechanism as mentioned in Equation 7. A Normalization layer with element-wise affine and a Dropout layer are employed around this sub-layer. On the other hand, the second sublayer, XLNet Feed Forward, is a fully connected feed-forward network, whose outputs are also of dimension 768, the same as the outputs of the embedding layers. Like the previous sublayers, the Feed Forward layer is also surrounded by a Normalization layer and a Dropout layer, however, another 2 Linear layers are added between them. Then, an additional Dropout layer is counted. It is notable that we only take the rate of 0.1 for every Dropout layer inside our

[2] https://fasttext.cc/
[3] https://github.com/flairNLP/flair
[4] https://github.com/stanfordnlp/stanza
[5] https://github.com/zihangdai/xlnet

model, from sublayers to inside sublayers. After 12 XLNet layers, another Dropout layer is added before the last Linear layer. We capture the intermediary output before the last Linear layer as the contextual features. Due to the complexity of XLNet, we also freeze some layers and keep only up to 4 layers to reduce the overfitting.

Proposed Model In order to maintain alignments between input tokens and their corresponding labels as well as to match corresponding representations from global features to contextual features in the same sentence additional steps were taken. First, we define an attention mask in XLNet as a sequence of 1s and 0s, with 1s for the first sub-word as the whole word embedding after tokenization and 0s for all padding sub-words. Then, in GCN features, we map the corresponding word representation at the position that the XLNet attention mark returns 1s and pad 0 otherwise. Therefore, each sentence has the same vector dimension in both global and contextual embeddings, which simplifies the concatenation.

In our implementation, we have used a GPU 2070 Super and a TitanX GPU with 56 CPUs and 128 GB RAM. The hyperparameters were 300 as embedding size, 16 as batch size, $5e-5$ as learning rate, 0.5 as dropout rate, 4 for number of epochs.

5 Results

We conducted multiple experiments to investigate the impact of global and contextual features on NER. More specifically, we implemented the architecture with only global features (given GloVe embeddings and spaCy universal dependencies GCN), only contextual features (given XLNet pretrained model), and then the proposed joint architecture combining both feature types.

As shown in Table 1b, the model achieves **93.82%** in F_1 score when we combine both global and contextual features, which outperforms the two variants using global or contextual features alone. In terms of recognition of specific entity types, the details are provided in Table 1a, showing that person named entities are the category where the best results are achieved, while the lowest results are with the MISC category, that is, the category of all named entities that do not belong to any of the predefined categories (location, organisation and person). Note also that in our model, using only training data and publicly available word embeddings (GloVe), our proposed model has competitive results without the need of adding any extra complex encoder-decoder layers.

Table 1: Results from the proposed joint architecture combining contextual and global features.

(a) Performance evaluation per entity types.

Entity types	**Precision**	**Recall**	F_1**-score**
LOC	94.15	93.53	93.83
MISC	81.33	81.89	81.62
ORG	88.97	92.29	90.60
PER	96.67	97.09	96.88

(b) Results of the proposed joint architecture compared to only contextual or only global features.

Embeddings	F_1 **scores**
Global features	88.63
Contextual features	93.28
Global + contextual features	**93.82**

Furthermore, the benefit of the proposed joint architecture is illustrated on the CoNLL 2003 example in Figure 2. While contextual features (XLNet), which are used in the majority of recent SOTA approaches, misclassifies the entity, the prediction from GCN and the

combined model correctly tags "MACEDONIA" as the name of a location, confirming our hypothesis on the effect of global features.

"... are the newcomers for the European group eight clash in Macedonia on December 14 ..."

Word	Predictions from GCNs	Predictions from XLNet	Predictions from combined model	Groundtruth
MACEDONIA	LOC	ORG	LOC	LOC

Fig. 2: GCN, XLNet, and our prediction on sample data.

Last but not least, in Table 2 we compare our results with reported SOTA results on the same dataset (we consider approaches since 2017). Note however, that some of the results are not directly comparable as in some works the final models are trained on both training and validation data, while we used training data only. It can be observed that our results outperform the SOTA approaches by a small margin (the current benchmark is 93.5 % F_1 score, compared to 93.82 % F_1 score achieved with our proposed approach). Furthermore, we notice that NER performance can be boosted with external knowledge (i.e. leveraging pretrained embeddings), as proven in our approach as well as in top benchmarks [26–28]. More importantly, complex decoder layers (CRF, Semi-CRF,...) do not always lead to better performance in comparison with softmax classification when we take advantage of contextualized language model embeddings.

6 Conclusion and future work

We propose a novel hierarchical neural model for NER that uses both the global features captured via graph representation and contextual features at the sentence level via XLNet pretrained model. The combination of global and contextual embeddings is proved to have a significant effect on the performance of NER tasks. Empirical studies on CoNLL 2003 English dataset suggest that our approach outperforms systems using only global or contextual features, and is competitive with SOTA methods (surpassing the current benchmarks by a small margin, with F_1 score up to 93.82 %). Given the promising results in English, our future work will consist of adapting the method to other languages, as well as cross-lingual experimental settings. In addition, we will consider further developing the method by incorporating also background knowledge from knowledge graphs and ontologies.

Work	**Input Representation**			**Context encoder**	**Context decoder**	**F-scores (%)**
	Char	**Word**	**Hybrid**			
Tran et al. [41]	LSTM	GloVe	-	LSTM	CRF	91.07
Ma and Hovy [29]	CNN	GloVe	-	LSTM	CRF	91.21
e and Ling [48]	LSTM	GloVe	-	LSTM	Semi-CRF	91.38
Yang et al. [44]	CNN	SENNA	-	LSTM	Reranker	91.62
Liu et al. [24]	LSTM	GloVe	-	LSTM	CRF	91.71
Peters et al. [34]	GRU	SENNA	LM	GRU	CRF	91.93
Peters et al. [35]	CNN-LSTM-LM	-	-	LSTM	CRF	92.22
Xia et al. [42]	LSTM	GloVe	ELMo, POS	LSTM	Softmax	92.28
Jie et al. [14]	-	GloVe	ELMo, dependency	LSTM	CRF	92.40
Liu et al. [26]	CNN	GloVe	ELMo, gazetteers	LSTM	Semi-CRF	92.75
Devlin et al. [7]	-	WordPiece	Segment, position	Transformer	Softmax	92.80
Li et al. [21]	-	-	BERT	-	Softmax	93.04
Yang et al. [46]	GRU	SENNA	-	GRU	CRF	93.09
Li et al. [22]	-	-	BERT	-	Softmax, Dice Loss	93.33
Luo et al. [28]	LSTM	GloVe	BERT, document-level embeddings	LSTM	CRF	93.37
Liu et al. [27]	CNN	GloVe	BERT, global embeddings	GRU	GRU	93.47
Jiang et al. [13]	-	GloVe	Pooled contextual embeddings	RNN	CRF	93.47
Baevski et al. [1]	CNN	-	Cloze-style LM embeddings	LSTM	CRF	**93.50**
Ours	-	GloVe	XLNet, global embeddings		Softmax	**93.82**

Table 2: Comparison of our proposal against SOTA techniques on the CoNLL 2003 dataset in terms of F_1 score. Values were taken from original papers and sorted by ascending order.

References

1. Baevski, A., Edunov, S., Liu, Y., Zettlemoyer, L., Auli, M.: Cloze-driven pretraining of self-attention networks. 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). pp. 5360–5369 (2019)
2. Cetoli, A., Bragaglia, S., O'Harney, A., Sloan, M.: Graph convolutional networks for named entity recognition. In: Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories. pp. 37–45 (2017)
3. Chieu, H.L., Ng, H.T.: Named entity recognition: a maximum entropy approach using global information. In: COLING 2002: The 19th International Conference on Computational Linguistics (2002)
4. Chiu, J.P., Nichols, E.: Named entity recognition with bidirectional lstm-cnns. Transactions of the Association for Computational Linguistics **4**, 357–370 (2016)
5. Collins, M., Singer, Y.: Unsupervised models for named entity classification. In: 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (1999)
6. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data (2018)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT (1) (2019)
8. Duvenaud, D.K., Maclaurin, D., Iparraguirre, J., Bombarell, R., Hirzel, T., Aspuru-Guzik, A., Adams, R.P.: Convolutional networks on graphs for learning molecular fingerprints. In: Advances in neural information processing systems. pp. 2224–2232 (2015)
9. Etzioni, O., Cafarella, M., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. Artificial intelligence **165**(1), 91–134 (2005)
10. Grishman, R., Sundheim, B.M.: Message understanding conference-6: A brief history. In: COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics (1996)
11. Hoffart, J., Yosef, M.A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust disambiguation of named entities in text. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. pp. 782–792 (2011)
12. Ji, Z., Sun, A., Cong, G., Han, J.: Joint recognition and linking of fine-grained locations from tweets. In: Proceedings of the 25th international conference on world wide web. pp. 1271–1281 (2016)
13. Jiang, Y., Hu, C., Xiao, T., Zhang, C., Zhu, J.: Improved differentiable architecture search for language modeling and 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). pp. 3576–3581 (2019)
14. Jie, Z., Lu, W.: 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). pp. 3853–3863 (2019)
15. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks (2019)
16. Krupka, G., IsoQuest, K.: Description of the nerowl extractor system as used for muc-7. In: Proceedings of the 7th Message Understanding Conference, Virginia. pp. 21–28 (2005)
17. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 260–270 (2016)
18. Lample, G., Conneau, A.: Cross-lingual language model pretraining. arXiv pp. arXiv–1901 (2019)
19. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: A lite bert for self-supervised learning of language representations. In: International Conference on Learning Representations (2019)

20. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering (2020)
21. Li, X., Feng, J., Meng, Y., Han, Q., Wu, F., Li, J.: A unified mrc framework for named entity recognition. arXiv pp. arXiv–1910 (2019)
22. Li, X., Sun, X., Meng, Y., Liang, J., Wu, F., Li, J.: Dice loss for data-imbalanced nlp tasks. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 465–476 (2020)
23. Liao, W., Veeramachaneni, S.: A simple semi-supervised algorithm for named entity recognition. In: Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing. pp. 58–65 (2009)
24. Liu, L., Shang, J., Ren, X., Xu, F.F., Gui, H., Peng, J., Han, J.: Empower sequence labeling with task-aware neural language model. In: 32nd AAAI Conference on Artificial Intelligence, AAAI 2018. pp. 5253–5260. AAAI Press (2018)
25. Liu, S., Sun, Y., Li, B., Wang, W., Zhao, X.: Hamner: Headword amplified multi-span distantly supervised method for domain specific named entity recognition. In: AAAI. pp. 8401–8408 (2020)
26. Liu, T., Yao, J.G., Lin, C.Y.: Towards improving neural named entity recognition with gazetteers. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 5301–5307 (2019)
27. Liu, Y., Meng, F., Zhang, J., Xu, J., Chen, Y., Zhou, J.: Gcdt: A global context enhanced deep transition architecture for sequence labeling. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 2431–2441 (2019)
28. Luo, Y., Xiao, F., Zhao, H.: Hierarchical contextualized representation for named entity recognition. In: AAAI. pp. 8441–8448 (2020)
29. Ma, X., Hovy, E.H.: End-to-end sequence labeling via bi-directional lstm-cnns-crf. In: ACL (1) (2016)
30. Mikheev, A., Moens, M., Grover, C.: Named entity recognition without gazetteers. In: Ninth Conference of the European Chapter of the Association for Computational Linguistics (1999)
31. Nadeau, D., Turney, P.D., Matwin, S.: Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In: Conference of the Canadian society for computational studies of intelligence. pp. 266–277. Springer (2006)
32. Palshikar, G.K.: Techniques for named entity recognition: a survey. In: Bioinformatics: Concepts, Methodologies, Tools, and Applications, pp. 400–426. IGI Global (2013)
33. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
34. Peters, M., Ammar, W., Bhagavatula, C., Power, R.: Semi-supervised sequence tagging with bidirectional language models. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1756–1765 (2017)
35. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of NAACL-HLT. pp. 2227–2237 (2018)
36. Ritter, A., Clark, S., Etzioni, O., et al.: Named entity recognition in tweets: an experimental study. In: Proceedings of the 2011 conference on empirical methods in natural language processing. pp. 1524–1534 (2011)
37. Rocktäschel, T., Weidlich, M., Leser, U.: Chemspot: a hybrid system for chemical named entity recognition. Bioinformatics **28**(12), 1633–1640 (2012)
38. Seti, X., Wumaier, A., Yibulayin, T., Paerhati, D., Wang, L., Saimaiti, A.: Named-entity recognition in sports field based on a character-level graph convolutional network. Information **11**(1), 30 (2020)
39. Subramanian, S., Trischler, A., Bengio, Y., Pal, C.J.: Learning general purpose distributed sentence representations via large scale multi-task learning. In: International Conference on Learning Representations (2018)
40. Tjong Kim Sang, E.F., De Meulder, F.: Introduction to the conll-2003 shared task: Language-independent named entity recognition. In: Daelemans, W., Osborne, M. (eds.) Proceedings of CoNLL-2003. pp. 142–147. Edmonton, Canada (2003)

41. Tran, Q.H., MacKinlay, A., Yepes, A.J.: Named entity recognition with stack residual lstm and trainable bias decoding. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 566–575 (2017)
42. Xia, C., Zhang, C., Yang, T., Li, Y., Du, N., Wu, X., Fan, W., Ma, F., Yu, P.: Multi-grained named entity recognition. In: 57th Annual Meeting of the Association for Computational Linguistics, ACL 2019. pp. 1430–1440. Association for Computational Linguistics (ACL) (2020)
43. Yadav, V., Bethard, S.: A survey on recent advances in named entity recognition from deep learning models. arXiv pp. arXiv–1910 (2019)
44. Yang, J., Zhang, Y., Dong, F.: Neural reranking for named entity recognition. In: Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017. pp. 784–792 (2017)
45. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: Xlnet: Generalized autoregressive pretraining for language understanding. In: Advances in neural information processing systems. pp. 5753–5763 (2019)
46. Yang, Z., Salakhutdinov, R., Cohen, W.: Multi-task cross-lingual sequence tagging from scratch. arXiv pp. arXiv–1603 (2016)
47. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 7370–7377 (2019)
48. Ye, Z., Ling, Z.H.: Hybrid semi-markov crf for neural sequence labeling. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 235–240 (2018)
49. Zhang, Y., Liu, Q., Song, L.: Sentence-state lstm for text representation. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 317–327 (2018)

F Slav-NER: the 3rd Cross-lingual Challenge on Recognition, Normalization, Classification, and Linking of Named Entities across Slavic Languages

Slav-NER: the 3rd Cross-lingual Challenge on Recognition, Normalization, Classification, and Linking of Named Entities across Slavic languages

Jakub Piskorski,[1] Bogdan Babych,[2] Zara Kancheva,[3] Olga Kanishcheva,[4] Maria Lebedeva,[5] Michał Marcińczuk,[6] Preslav Nakov,[7] Petya Osenova,[3] Lidia Pivovarova,[8] Senja Pollak,[9] Pavel Přibáň,[10] Ivaylo Radev,[3] Marko Robnik-Šikonja,[11] Vasyl Starko,[12] Josef Steinberger,[10] Roman Yangarber[8]

[1]Polish Academy of Sciences, Warsaw, Poland jpiskorski@gmail.com
[2]Heidelberg University, Germany bogdan.babych@iued.uni-heidelberg.de
[3]IICT-BAS, Bulgarian Academy of Sciences, Bulgaria {petya,zara,radev}@bultreebank.org
[4]Kharkiv Polytechnic Institute, Ukraine kanichshevaolga@gmail.com
[5]Pushkin State Russian Language Institute, Moscow, Russia mylebedeva@pushkin.institute
[6]Wrocław University of Science and Technology, Poland michal.marcinczuk@pwr.edu.pl
[7]Qatar Computing Research Institute, HBKU pnakov@hbku.edu.qa
[8]University of Helsinki, Finland first.last@helsinki.fi
[9]Jozef Stefan Institute, Slovenia senja.pollak@ijs.si
[10]University of West Bohemia, Czech Republic {pribanp,jstein}@kiv.zcu.cz
[11]University of Ljubljana, Slovenia Marko.RobnikSikonja@fri.uni-lj.si
[12]Ukrainian Catholic University, Lviv, Ukraine vstarko@gmail.com

Abstract

This paper describes Slav-NER: the 3rd Multilingual Named Entity Challenge in Slavic languages. The tasks involve recognizing mentions of named entities in Web documents, normalization of the names, and cross-lingual linking. The Challenge covers six languages and five entity types, and is organized as part of the 8^{th} Balto-Slavic Natural Language Processing Workshop, co-located with the EACL 2021 Conference. Ten teams participated in the competition. Performance for the named entity recognition task reached 90% F-measure, much higher than reported in the first edition of the Challenge. Seven teams covered all six languages. Detailed evaluation information is available on the shared task web page.

1 Introduction

Analyzing named entities (NEs) in Slavic languages poses a challenging problem, due to the rich inflection and derivation, free word order, and other morphological and syntactic phenomena exhibited in these languages (Przepiórkowski, 2007; Piskorski et al., 2009). Encouraging research on detection and normalization of NEs—and on the closely related problem of cross-lingual, cross-document *entity linking*—is of paramount importance for improving multilingual and cross-lingual information access in these languages.

This paper describes the 3^{rd} Shared Task on multilingual NE recognition (NER), which aims at addressing these problems in a systematic way. The shared task was organized in the context of the 8^{th} BSNLP: Balto-Slavic Natural Language Processing Workshop, co-located with the EACL 2021 conference. The task covers six languages—Bulgarian, Czech, Polish, Russian, Slovene and Ukrainian—and five types of NE: person, location, organization, product, and event. The input text collection consists of documents collected from the Web, each collection centered on a certain "focal" event. The rationale of such a setup is to foster the development of "end-to-end" NER and cross-lingual entity linking solutions, which are not tailored to specific, narrow domains. This paper also serves as an introduction and a guide for researchers wishing to explore these problems using the training and test data, which are released to the public.[1]

The paper is organized as follows. Section 2 reviews prior work. Section 3 describes the task; Section 4 describes the annotation of the dataset. The evaluation methodology is introduced in Section 5. Participant systems are described in Section 6, and the results obtained by these systems are presented in Section 7. We present the conclusions and lessons learned in Section 8.

2 Prior Work

The work described here builds on the 1^{st} and 2^{nd} Shared Task on Multilingual Named Entity Recognition, Normalization and cross-lingual Match-

[1]bsnlp.cs.helsinki.fi/shared_task.html

Proceedings of the 8th BSNLP Workshop on Balto-Slavic Natural Language Processing, pages 122–133

ing for Slavic Languages, (Piskorski et al., 2017, 2019), which, to the best of our knowledge, are the first attempts at such shared tasks covering multiple Slavic languages.

High-quality recognition and analysis of NEs is an essential step not only for information access, such as document retrieval and clustering, but it also constitutes a fundamental processing step in a wide range of NLP pipelines built for higher-level analysis of text, such as Information Extraction, see, e.g. (Huttunen et al., 2002). Other NER-related shared tasks have been organized previously. The first *non-English* monolingual NER evaluations—covering Chinese, Japanese, Spanish, and Arabic—were held in the context of the Message Understanding Conferences (MUCs) (Chinchor, 1998) and the ACE Programme (Doddington et al., 2004). The first *multilingual* NER shared task, which covered several European languages, including Spanish, German, and Dutch, was organized in the context of the CoNLL conferences (Tjong Kim Sang, 2002; Tjong Kim Sang and De Meulder, 2003). The NE types covered in these campaigns were similar to the NE types covered in our Challenge. Worth mentioning in this context is Entity Discovery and Linking (EDL) (Ji et al., 2014, 2015), a track of the NIST Text Analysis Conferences (TAC). EDL aimed to extract entity mentions from a collection of documents in multiple languages (English, Chinese, and Spanish), and to partition the entities into cross-document equivalence classes, by either linking mentions to a knowledge base or directly clustering them. An important difference between EDL and our task is that EDL required linking entities to a pre-existing knowledge base.

Related to cross-lingual NE recognition is NE transliteration, i.e., linking NEs across languages that use different scripts. A series of NE Transliteration Shared Tasks were organized as a part of NEWS—Named Entity Workshops—(Duan et al., 2016), focusing mostly on Indian and Asian languages. In 2010, the NEWS Workshop included a shared task on Transliteration Mining (Kumaran et al., 2010), i.e., mining of names from parallel corpora. This task included corpora in English, Chinese, Tamil, Russian, and Arabic.

Research on NE focusing on Slavic languages includes tools for NE recognition for Croatian (Karan et al., 2013; Ljubešić et al., 2013), NE recognition in Croatian tweets (Baksa et al., 2017), a manually annotated NE corpus for Croatian (Agić and Ljubešić, 2014), tools for NE recognition in Slovene (Štajner et al., 2013; Ljubešić et al., 2013), a Czech corpus of 11K annotated NEs (Ševčíková et al., 2007), NER tools for Czech (Konkol and Konopík, 2013), tools and resources for fine-grained annotation of NEs in the National Corpus of Polish (Waszczuk et al., 2010; Savary and Piskorski, 2011), NER shared tasks for Polish organized under the umbrella of POLEVAL[2] evaluation campaigns (Ogrodniczuk and Łukasz Kobyliński, 2018, 2020). and a recent shared task on NE Recognition in Russian (Starostin et al., 2016).

3 Task Description

The data for this edition of the shared task consists of sets of documents in six Slavic languages: Bulgarian, Czech, Polish, Russian, Slovene and Ukrainian. To accommodate entity linking, each set of documents is chosen to revolve around one certain entity—e.g., a person, an organization or an event. The documents were obtained from the Web, by posing a keyword query to a search engine or publicly available crawled data repositories, and extracting the textual content from the respective sources.

The task is to recognize, classify, and "normalize" all named-entity mentions in each of the documents, and to link across languages all named mentions referring to the same real-world entity. Formally, the Multilingual Named Entity Recognition task is subdivided into three sub-tasks:

- **Named Entity Mention Detection and Classification:** Recognizing all named mentions of entities of five types: persons (PER), organizations (ORG), locations (LOC), products (PRO), and events (EVT).

- **Name Normalization:** Mapping each named mention of an entity to its corresponding *base form*. By "base form" we generally mean the lemma ("dictionary form") of the inflected word-form. In some cases normalization should go beyond inflection and transform a derived word into a base word's lemma, e.g., in case of personal possessives (see below). Multi-word names should be normalized to the *canonical multi-word expression*—rather than a sequence

[2] http:\\poleval.pl

of lemmas of the words making up the multi-word expression.

- **Entity Linking.** Assigning a unique identifier (ID) to each detected named mention of an entity, in such a way that mentions referring to the same real-world entity should be assigned the same ID—referred to as the cross-lingual ID.

The task does not require positional information of the name entity mentions. Thus, for all occurrences of the same form of a NE mention (e.g., an inflected variant, an acronym or abbreviation) within a given document, no more than one annotation should be produced.[3] Furthermore, distinguishing typographical case is not necessary since the evaluation is case-insensitive. If the text includes lowercase, uppercase or mixed-case variants of the same entity, the system should produce only one annotation for all of these mentions. For instance, for "*ISIS*" and "*isis*" (provided that they refer to the same NE type), only one annotation should be produced. The recognition of common-noun or pronominal references to named entities does not constitute part of the task.

3.1 Named Entity Classes

The task defines the following five NE classes.

Person names (PER): Names of real (or fictional) persons). Person names should not include titles, honorifics, and functions/positions. For example, in the text fragment "*... President Vladimir Putin...*", only "*Vladimir Putin*" is recognized as a person name. Both initials and pseudonyms are also considered named mentions of persons. Similarly, toponym-based named references to groups of people (that do not have a formal organization unifying them) should also be recognized, e.g., "*Germans*." In this context, mentions of a single member belonging to such groups, e.g., "*German*," should be assigned the same cross-lingual ID as plural mentions, i.e., "*Germans*" and "*German*" when referring to the nation receive the same cross-lingual ID.

Named mentions of other groups of people that do have a formal organization unifying them should be tagged as PER, e.g., in the phrase "*Spart'ané vyhráli*" (Spartans won), "*Spart'ané* are to be tagged as PER.

Personal possessives derived from a person's name should be classified as a Person, and the base form of the corresponding name should be extracted. For instance, in "*Trumpov tweet*" (Croatian) one is expected to classify "*Trumpov*" as PER, with the base form "*Trump*."

Locations (LOC): All toponyms and geopolitical entities—cities, counties, provinces, countries, regions, bodies of water, land formations, etc.—including named mentions of *facilities*—e.g., stadiums, parks, museums, theaters, hotels, hospitals, transportation hubs, churches, streets, railroads, bridges, and similar facilities.

In case named mentions of facilities *also* refer to an organization, the LOC tag should be used. For example, from the text "*San Rafaelle Hospital hired new staff due to Covid-19 pandemic*" the mention "*San Rafaelle Hospital*" should be classified as LOC.

Organizations (ORG): All organizations, including companies, public institutions, political parties, international organizations, religious organizations, sport organizations, educational and research institutions, etc.

Organization designators and potential mentions of the seat of the organization are considered to be part of the organization name. For instance, from the text "*...Zakład Ubezpieczeń Społecznych w Bydgoszczy...*" (The Social Insurance Institution in Bydgoszcz), the full phrase "*Zakład Ubezpieczeń Społecznych w Bydgoszczy*" should be extracted.

Products (PRO): All names of *products and services*, such as electronics ("*Samsung Galaxy A41*"), cars ("*Honda Pilot*"), newspapers ("*Der Spiegel*"), web-services ("*Pintertest*"), medicines ("*Oxycodone*"), awards ("*Pulitzer Prize*"), books ("*Animal Farm*"), TV programmes ("*Wiadomości TVP*"), etc.

When a company name is used to refer to a *service*, e.g., "*na Instagramie*" (Polish for "on Instagram"), the mention of "*Instagramie*" is considered to refer to a service/product and should be tagged as PRO. However, when a company name refers to a service, expressing an opinion of the company, it should be tagged as ORG.

This category also includes legal documents and treaties, e.g., "*Układ z Schengen*" (Pol-

[3] Unless the different occurrences have different entity types (different *readings*) assigned to them, which is rare.

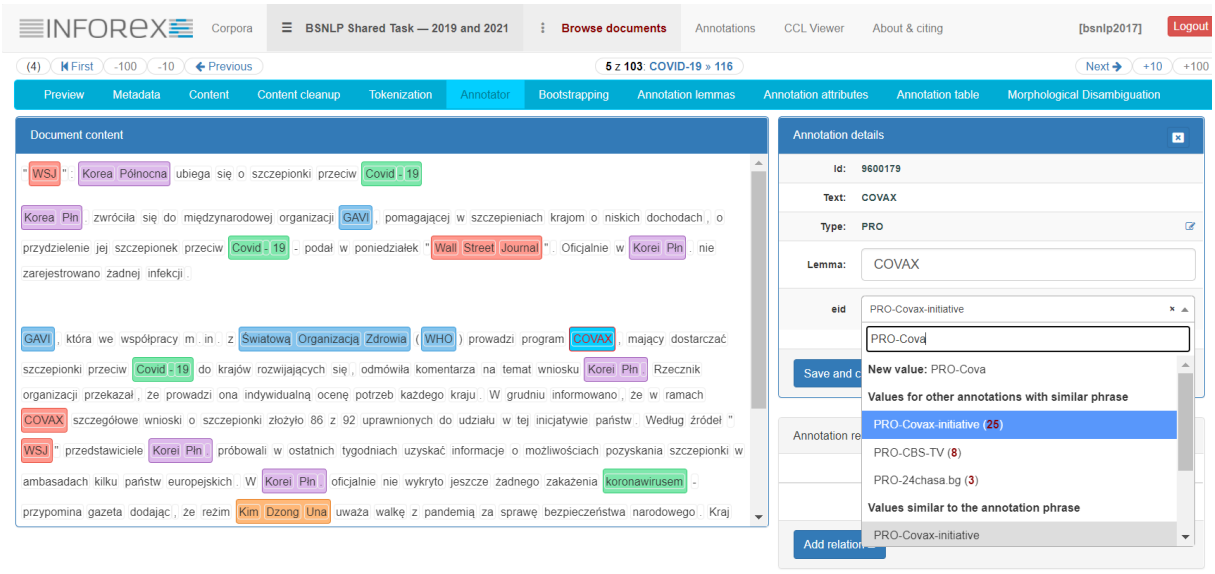


Figure 1: Screenshot of the Inforex Web interface, the tool used for data annotation.

Jacob Serrano (23) z americké Floridy se stal vůbec prvním Američanem, který byl oočkován experimentální vakcínou proti koronaviru, ta vznikla za spolupráce vědců z Oxfordské univerzity a farmaceutické společnosti AstraZeneca. Podle WHO jde zatím o nejslibnější očkovací látku. Serrano se neváhal zapojit se do boje s koronavirem, který způsobuje nemoc covid-19, nákaza ho totiž připravila o 7 příbuzných, uvedl list Daily Mail.

114

Američanem	Američan	PER	GPE-USA
AstraZeneca	AstraZeneca	ORG	ORG-AstraZeneca
Daily Mail	Daily Mail	PRO	PRO-Daily-Mail
Floridy	Florida	LOC	GPE-Florida
Jacob Serrano	Jacob Serrano	PER	PER-Jacob-Serrano
Oxfordské univerzity	Oxfordská univerzita	ORG	ORG-University-of-Oxford
Serrano	Serrano	PER	PER-Jacob-Serrano
WHO	WHO	ORG	ORG-World-Health-Org
covid-19	covid-19	EVT	EVT-Covid-19
koronavirem	koronavirus	EVT	EVT-Covid-19
koronaviru	koronavirus	EVT	EVT-Covid-19

Figure 2: Example input and output formats.

ish: "Schengen Agreement") and initiatives, e.g., "*Horizon 2020*".

Events (EVT): This category covers named mentions of events, including conferences, e.g. "24. Konference Žárovného Zinkování" (Czech: "Hot Galvanizing Conference"), concerts, festivals, holidays, e.g., "*Święta Bożego Narodzenia*" (Polish: "Christmas"), wars, battles, disasters, e.g., "*Katastrofa Smoleńska*" (Polish: "the Smoleńsk air disaster"), outbreaks of infectious diseases ("*Spanish Flu*"). Future, speculative, and fictive events—e.g., "'Polexit"—are considered event mentions too.

3.2 Complex and Ambiguous Entities

In case of complex named entities, consisting of nested named entities, only the *top-most* entity should be recognized. For example, from the text "*Università Commerciale Luigi Bocconi*" one should not extract "*Luigi Bocconi*", but only the top-level entity.

In case one word-form (e.g., "*Georgia*") is used to refer to more than one different real-world entities in different contexts in the same document (e.g., a person and a location), two annotations should be returned, associated with different cross-lingual IDs.

In case of coordinated phrases, like "*European and German Parliament*," two names should be extracted (as ORG). The lemmas would be "*European*" and "*German Parliament*", and the IDs should refer to "*European Parliament*" and "*German Parliament*" respectively.

In rare cases, plural forms might have two annotations—e.g., in the phrase "*a border between Irelands*"—"*Irelands*" should be extracted twice with identical lemmas but different IDs.

3.3 System Input and Response

Input Document Format: Documents in the collection are represented in the following format. The first five lines contain the following metadata (in the respective order): `<DOCUMENT-ID>`,

<LANGUAGE>, <CREATION-DATE>, <URL>, <TITLE>, <TEXT>. The text to be processed begins from the sixth line and runs till the end of file. The <URL> field stores the origin from which the text document was retrieved. The values of <CREATION-DATE> and <TITLE> were not provided for all documents, due to unavailability of such data or due to errors in parsing during data collection.

System Response. For each input file, the system should return one output file as follows. The first line should contain only the <DOCUMENT-ID>, which corresponds to the input. Each subsequent line contains one annotation, as tab-separated fields:

<MENTION> TAB <BASE> TAB <CAT> TAB <ID>

The <MENTION> field should be the NE as it appears in text. The <BASE> field should be the base form of the entity. The <CAT> field stores the category of the entity (ORG, PER, LOC, PRO, or EVT) and <ID> is the cross-lingual identifier. The cross-lingual identifiers may consist of an arbitrary sequence of alphanumeric characters. An example document in Czech and the corresponding response is shown in Figure 2.

The detailed descriptions of the tasks are available on the web page of the Shared Task.[4]

4 Data

For Russian, Polish, Czech and Bulgarian, the training and test data sets from the 2019 Shared Task were used as training data for 2021. For the new languages—Ukrainian and Slovene—new training sets were annotated. The test data in all six languages covered two major current topics: the COVID-19 pandemic and the 2020 USA Presidential elections (USA 2020 ELECTIONS).

The 2019 training data consist of four sets of documents extracted from the Web, each related to a given *focus* entity. We tried to choose entities related to events in 2018 and 2019 covered in mainstream news in many languages. ASIA BIBI, which relates to a Pakistani woman involved in a blasphemy case, BREXIT, RYANAIR, which faced a massive strike, and NORD STREAM, a controversial Russian-European project.

Each dataset was created as follows. For the focus entity, we posed a search query to Google and/or publicly available crawled data repositories, in each of the target languages. The query returned documents in the target language. We removed duplicates, downloaded the HTML—mainly news articles—and converted them into plain text. Since the result of HTML parsing may include not only the main text of a Web page, but also spurious text, some additional manual cleaning was applied whenever necessary. The resulting set of "cleaned" documents were used to manually select documents for each language and topic, for the final datasets.

Documents were annotated using the Inforex[5] web-based system for annotation of text corpora (Marcińczuk et al., 2017). Inforex allows parallel access and resource sharing by multiple annotators. It let us share a common list of entities, and perform entity-linking semi-automatically: for a given entity, an annotator sees a list of entities of the same type inserted by all annotators and can select an entity ID from the list. A snapshot of the Inforex interface is in Figure 1.

In addition, Inforex keeps track of all lemmas and IDs inserted for each surface form, and inserts them automatically, so in many cases the annotator only confirms the proposed values, which speeds up the annotation process a great deal. All annotations were made by native speakers. After annotation, we performed automatic and manual consistency checks, to reduce annotation errors, especially in entity linking.

Training and test data statistics are presented in Table 1 and 2 respectively.

The testing datasets—COVID-19 and USA 2020 ELECTIONS—were released to the participants who were given circa 2 days to return up to 5 system responses. The participants did not know the topics in advance, and did not receive the annotations. The main drive behind this decision was to push participants to build a general solution for Slavic NER, rather than to optimize their models toward a particular set of names.

5 Evaluation Methodology

The NER task (exact case-insensitive matching) and Name Normalization (or "lemmatization") were evaluated in terms of precision, recall, and F1-measure. For NER, two types of evaluations were carried out:

[4] http://bsnlp.cs.helsinki.fi/System_response_guidelines-1.2.pdf

[5] github.com/CLARIN-PL/Inforex

	Brexit						Asia Bibi						Nord Stream						Ryanair					
	PL	CS	RU	BG	SL	UK	PL	CS	RU	BG	SL	UK	PL	CS	RU	BG	SL	UK	PL	CS	RU	BG	SL	UK
Documents	500	284	153	600	52	50	88	89	118	101	4	6	151	161	150	130	74	40	146	163	150	87	52	63
PER	2 650	1 108	1 308	2 515	532	242	683	570	643	583	36	39	538	570	392	335	548	78	136	161	72	147	107	33
LOC	3 524	1 279	666	2 407	403	336	403	366	567	388	24	57	1 430	1 689	1 320	910	1 362	339	821	871	902	344	384	455
ORG	3 080	1 039	828	2 455	301	166	286	214	419	245	10	30	837	477	792	540	460	449	529	707	500	238	408	193
EVT	1 072	471	261	776	165	62	14	3	1	8	0	0	15	9	5	6	50	14	7	12	0	4	8	0
PRO	668	232	137	490	31	17	55	42	49	63	2	1	405	364	510	331	243	8	114	66	82	79	101	20
Total	10 994	4 129	3 200	8 643	1 445	823	1 441	1 195	1 679	1 287	72	127	3 225	3 116	3 020	2 122	2 664	948	1 607	1 817	1 556	812	1008	701
Distinct																								
Surface forms	2 820	1 111	783	1 200	596	234	508	303	406	412	51	87	845	770	892	504	902	336	514	475	400	323	673	187
Lemmas	2 133	840	568	1 091	411	177	412	248	317	360	41	77	634	550	583	448	600	244	419	400	332	315	520	137
Entity IDs	1 506	583	268	772	288	127	273	160	178	230	31	64	441	392	321	305	465	177	322	306	251	245	428	108

Table 1: Overview of the training datasets.

	Covid-19						USA 2020 Elections					
	PL	CS	RU	BG	SL	UK	PL	CS	RU	BG	SL	UK
Documents	103	155	83	151	178	85	66	85	163	151	143	83
PER	419	478	559	351	834	215	566	447	3203	1539	2589	672
LOC	369	474	701	759	1228	364	827	277	3457	1093	1268	541
ORG	402	318	628	589	965	455	243	99	2486	557	578	384
EVT	240	393	435	465	612	269	86	63	396	170	118	257
PRO	137	155	400	168	274	143	87	56	846	240	254	124
Total	1567	1818	2723	2332	3913	1446	1810	942	10398	3599	4807	1978
Distinct												
Surface forms	688	941	1436	1092	2190	622	484	377	3440	1117	1605	537
Lemmas	557	745	1133	1016	1774	509	356	279	2593	1019	1129	390
Entity IDs	404	562	796	764	1400	369	278	200	1669	668	833	270

Table 2: Overview of the test datasets.

- **Relaxed:** An entity mentioned in a given document is considered to be extracted correctly if the system response includes *at least one* annotation of a named mention of this entity (regardless of whether the extracted mention is in base form);
- **Strict:** The system response should include exactly one annotation *for each* unique form of a named mention of an entity in a given document, i.e., identifying all variants of an entity is required.

In relaxed evaluation we additionally distinguish between *exact* and *partial matching*: in the latter case, an entity mentioned in a given document is considered to be extracted correctly if the system response includes at least one partial match of a named mention of this entity.

We evaluate systems at several levels of granularity: we measure performance for (a) all NE types and all languages, (b) each given NE type and all languages, (c) all NE types for each language, and (d) each given NE type per language.

In the name normalization task, we take into account only correctly recognized entity mentions and only those that were normalized (on both the annotation and system's sides). Formally, let $N_{correct}$ denote the number of all correctly recognized entity mentions for which the system returned a correct base form. Let N_{key} denote the number of all normalized entity mentions in the gold-standard answer key and $N_{response}$ denote the number of all normalized entity mentions in the system's response. We define precision and recall for the name normalization task as:

$$Recall = \frac{N_{corrrect}}{N_{key}} \qquad Precision = \frac{N_{corrrect}}{N_{response}}$$

In evaluating document-level, single-language and cross-lingual entity linking we adopted the Link-Based Entity-Aware metric (LEA) (Moosavi and Strube, 2016), which considers how important the entity is and how well it is resolved. LEA is defined as follows. Let $K = \{k_1, k_2, \ldots, k_{|K|}\}$ denote the set of key entities and $R = \{r_1, r_2, \ldots, r_{|R|}\}$ the set of response entities, i.e., $k_i \in K$ ($r_i \in R$) stand for set of mentions of the same entity in the key entity set (response entity set). LEA recall and precision are then defined as follows:

$$Recall_{LEA} = \frac{\sum_{k_i \in K} \left(imp(k_i) \cdot res(k_i) \right)}{\sum_{k_z \in K} imp(k_z)}$$

$$Precision_{LEA} = \frac{\sum_{r_i \in R} \left(imp(r_i) \cdot res(r_i) \right)}{\sum_{r_z \in R} imp(r_z)}$$

where imp and res denote the measure of importance and the resolution score for an entity, respectively. In our setting, we define $imp(e) = \log_2 |e|$ for an entity e (in K or R), $|e|$ is the number of mentions of e—i.e., the more mentions an entity has the more important it is. To avoid biasing the importance of the more frequent entities log

is used. The resolution score of key entity k_i is computed as the fraction of correctly resolved co-reference links of k_i:

$$res(k_i) = \sum_{r_j \in R} \frac{link(k_i \cap r_j)}{link(k_i)}$$

where $link(e) = (|e| \times (|e| - 1))/2$ is the number of unique co-reference links in e. For each k_i, LEA checks all response entities to check whether they are partial matches for k_i. Analogously, the resolution score of response entity r_i is computed as the fraction of co-reference links in r_i that are extracted correctly:

$$res(r_i) = \sum_{k_j \in K} \frac{link(r_i \cap k_j)}{link(r_i)}$$

LEA brings several benefits. For example, LEA considers resolved co-reference relations instead of resolved mentions and has more discriminative power than other metrics for co-reference resolution (Moosavi and Strube, 2016).

The evaluation was carried out in "case-insensitive" mode: all named mentions in system response and test corpora were lower-cased.

6 Participant Systems

Six teams submitted descriptions of their systems as BSNLP Workshop papers. We briefly review these systems here; for complete descriptions, please see the corresponding papers. Two additional teams submitted their results with short descriptions of their systems, which appear in this section.

The **UL FRI** system, (Prelevikj and Zitnik, 2021), generated results for several settings, models and languages, although the team's main motivation is to develop effective NER tools for Slovenian. The system uses contemporary BERT and RoBERTa multilingual pre-trained models, which include Slovene among other languages. The system was further trained on the SlavNER dataset for the NER task and used the Dedupe method for the Entity Matching task. The best performing models were pre-trained on Slovene. The results also indicate that two-step prediction of NE could be beneficial. The team made their code publicly available.

The **Priberam Labs** system, (Ferreira et al., 2021), focuses on the NER task. It uses three components: a multilingual contextual embedding model, a character-level embedding model, and a bi-affine classifier model. The paper reports results for different multilingual contextual embedding models, which included Multilingual BERT, XLM-RoBERTa, or the Slavic BERT. For different languages the best-performing models where different, but having the same language within the large pre-trained model usually improved the results—e.g., Slavic BERT, which used additional resources for Bulgarian, Russian and Polish, also performed best for these languages. The system uses heuristics to predict and resolve spans of NEs, and in this way it is able to tag overlapping entities. The code for the system is made available.

The **TLD** system, (Vīksna and Skadina, 2021), uses a staged approach. The first stage is identification of NEs in context, which is treated as a sequence labeling problem and is performed by a multilingual BERT model from Google, modified by the team. Entity linking is the second stage, which uses a list of LaBSE embeddings; matched entries need to pass a pre-defined threshold of cosine similarity with existing entries; otherwise they are added as new values to the list. The third stage is normalisation of identified entities, which is performed using models provided with Stanza.

The **L3i** system, (Cabrera-Diego et al., 2021), combines BERT models with the "Frustratingly Easy" domain adaptation algorithm. It also uses other techniques to improve system's NER performance, such as marking and enrichment of uppercase tokens, prediction of NE boundaries with a multitask approach, prediction of masked tokens, fine-tuning the language model to the domain of the document.

The **TraSpaS** system, (Suppa and Jariabka, 2021), tests the assumption that the universal open-source NLP toolkits (such as SpaCy, Stanza or Trankit) could achieve competitive performance on the Multilingual NER task, using large pre-trained Transformer-based language models available from HuggingfaceTransformers, which have not been available in previous editions of the Shared Task. The team tests the generalizability of the models to new low-resourced domains, and to languages such as Slovene and Ukrainian.

The **UWr-VL** system, (Rychlikowski et al., 2021), utilizes large collections of unstructured and structured documents for unsupervised training of embedding of lexical units and for recog-

nizing and linking multiple real-world NEs. In particular, the team makes use of CommonCrawl news articles, Wikipedia, and its structured counterpart Wikidata as knowledge sources, to address the problem of data scarcity, building neural gazetteer via collecting different embeddings from these knowledge sources. The system further uses standard neural approaches to the NER task, with a RNN classifier, in order to determine for every input word the probability of labelling it with various beginning and end NE tags.

Two more systems generated the results for the shared task—**CTC-NER** from the Cognitive Technologies Center team, and **PAISC_wxd**:

CTC-NER is a baseline prototype of a NER component of an entity recognition system currently under development at the Cognitive Technologies Center. The system has a hybrid architecture combining rule-based and ML techniques; the ML-component is loosely related to (Antonova and Soloviev, 2013). The languages currently processed include Russian, English and Ukrainian.

PAISC_wxd uses the XLM-Roberta model, followed by BiLSTM-CRF on top. In addition, the system uses data enhancement based on machine translation.

7 Evaluation Results

Figure 3 shows the performance of the systems averaged across all languages and both test corpora. For each team that provided a solution for all six languages (7 teams except **CTC-NER**), we present the best scores ($F1$, $Precision$, and $Recall$) obtained by the team in three evaluation modes.[6]

As the plots show, the best performing model, Priberam, yields F-measure 85.7% according to the *relaxed partial* evaluation, and 79.3% according to the strict evaluation. The Priberam submission scores highest in precision — 89,4% relaxed partial, and 85.1% strict — but much lower in recall — 82.2% relaxed partial, and 74.3% strict.

Among the teams that submitted results for *cross-lingual entity linking*, only two achieved results comparable with the benchmarks achieved on the Second Challenge, and this year's results surpass those benchmarks by a substantial margin. The best results for each team, averaged across two corpora, are shown in Table 3. These results

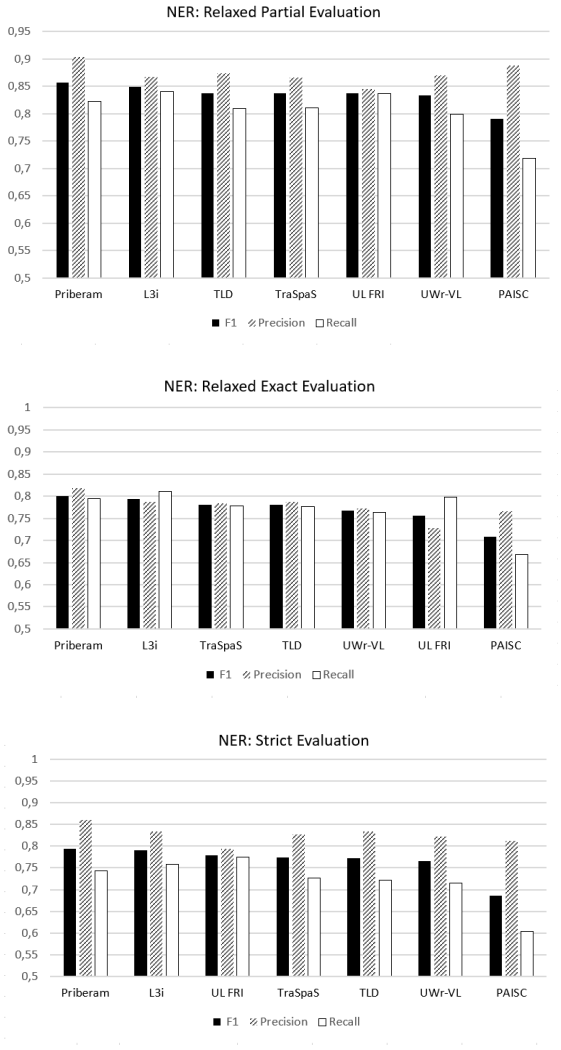


Figure 3: Best average performance scores obtained by the teams on the two test data

show that this task is much more difficult than entity extraction. The best performing model, TLD, achieves F-measure 50.4%.

Note that in our setting the performance on entity linking depends on the performance on name recognition and normalization: each system had to link entities that it had extracted from documents upstream, rather than link a set of correct entities.

Tables 4 and 5 present the F1-measures separated by language, for all tasks for the COVID-19 and USA 2020 ELECTIONS data sets These tables show only the top-performing model for each team. For recognition, we show only the *relaxed* evaluation, since the results obtained on the three evaluation schemes are correlated, as can be seen from Figure 3.

The tables indicate some variation in scores obtained on the test corpora This variation could be

[6]Complete results available on the Workshop's Web page: bsnlp.cs.helsinki.fi/final-rank-2021.pdf

COVID-19		USA 2020 ELECTIONS	
System	F1	System	F1
TLD	47.5	TLD	52.0
UWr-VL	32.8	UWr-VL	27.9
Priberam	5.8	Priberam	8.0
L3i	4.4	TraSpaS	7.9
PAISC	2.8	L3i	7.3
TraSpaS	2.7	PAISC	6.2
UL FRI	1.9	CTC-NER	2.9
CTC-NER	1.2	UL FRI	0.4

Table 3: Cross-lingual entity linking.

due to a number of factors, including actual differences in the test data, as well as differences in annotation across languages. This variation should and will be investigated in greater depth.

In Table 6 we present the results of the evaluation by entity type. As seen in the table, performance was higher overall for LOC and PER, and substantially lower for ORG and PRO, which corresponds with our findings from the previous editions of the shared task, where ORG and MISC were the most problematic categories (Piskorski et al., 2017). The PRO category also exhibits higher variance across languages and corpora than other categories, which might point to possible annotation artefacts. The results for the EVT category are less informative, since the task heavily depends on detecting the repeated central events of the corpora.

8 Conclusion

This paper reports on the 3^{rd} Multilingual Named Entity Challenge focusing on recognizing mentions of NEs in Web documents in six Slavic languages, normalization of the NEs, and cross-lingual entity linking. The Challenge has attracted substantial interest, following the prior Challenges in 2017 and 2019, with 10 teams registering for the competition and eight teams submitting results from working systems, with multiple variants. Most systems use state-of-the-art neural network models. Overall, the results of the best-performing systems are quite strong for extraction and normalization, while cross-lingual linking is the most challenging of the tasks.

We show summary results for the main aspects of the challenge and the best-performing model for each team. For detailed, in-depth evaluations of all participating systems and their performance please consult the Shared Task's Web page and the papers by the respective teams.

To stimulate further research into NLP for Slavic languages, including cross-lingual entity linking, our training and test datasets, the detailed annotations, and scripts used for evaluations are made available to the public on the Shared Task's Web page.[7] The annotation interface is released by the Inforex team, to support further annotation of additional data for future tests.

This challenge covered six Slavic languages. For future editions of the Challenge, we plan to expand the data sets, covering a wider range of entity types, and supporting cross-lingual entity linking. We plan to expand the training and test data to include *non-Slavic* languages. We will also undertake further refinement of the underlying annotation guidelines—a highly complex task in a real-world setting. More complex phenomena also need to be addressed, e.g., coordinated NEs, contracted versions of multiple NEs, etc.

We believe that the reported results and the annotated datasets will help stimulate further research on robust, end-to-end analysis of real-world texts in Slavic languages.

Acknowledgments

Work on Bulgarian was in part supported by the Bulgarian National Interdisciplinary Research e-Infrastructure for Resources and Technologies for the Bulgarian Language and Cultural Heritage, part of the EU infrastructures CLARIN and DARIAH – CLaDA-BG, Grant number DO1-377/18.12.2020.

Work on Czech was in part supported by ERDF "Research and Development of Intelligent Components of Advanced Technologies for the Pilsen Metropolitan Area (InteCom)" (no. CZ.02.1.01/0.0/0.0/17 048/0007267), and by Grant No. SGS-2019-018 "Processing of heterogeneous data and its specialized applications."

Work on Inforex and on Polish was supported in part by investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education.

We thank the students of Pushkin State Russian Language Institute for their assistance with annotation of Russian data.

This work has been partially supported by the European Union Horizon 2020 research and in-

[7] bsnlp.cs.helsinki.fi/shared_task.html

COVID-19		*Language*											
Phase	*Metric*	bg		cs		pl		ru		sl		uk	
Recognition	Relaxed	Priberam	83.2	UWr-VL	86.7	Priberam	87.8	L3i	76.0	UWr-VL	87.6	UWr-VL	84.8
	Partial	L3i	82.8	Priberam	86.3	UWr-VL	86.9	Priberam	75.1	Priberam	87.5	L3i	80.6
		TLD	82.2	TLD	84.1	TLD	86.4	PAISC	74.4	L3i	85.6	TLD	80.1
		UL FRI	81.6	L3i	83.9	L3i	85.0	TLD	72.9	TLD	84.2	Priberam	79.9
		UWr-VL	81.2	TraSpaS	82.0	UL FRI	83.4	UL FRI	71.9	TraSpaS	83.9	PAISC	78.3
		TraSpaS	80.9	UL FRI	80.4	TraSpaS	82.5	TraSpaS	70.2	PAISC	80.1	UL FRI	78.3
		PAISC	79.7	PAISC	77.6	PAISC	81.0	CTC-NER	69.3	UL FRI	79.1	TraSpaS	78.1
								UWr-VL	67.1			CTC-NER	65.0
Normalization		UWr-VL	33.3	TraSpaS	47.0	UWr-VL	57.4	CTC-NER	40.4	UWr-VL	53.0	TraSpaS	53.7
		UL FRI	21.4	TLD	45.2	UL FRI	47.2	UL FRI	39.9	UL FRI	40.5	UWr-VL	51.5
		TLD	13.8	UWr-VL	44.8	TraSpaS	46.2	TraSpaS	38.6	TraSpaS	34.3	UL FRI	50.7
		TraSpaS	10.0	UL FRI	44.4	TLD	45.3	TLD	36.2	TLD	32.3	TLD	46.3
		Priberam	0.0	Priberam	0.0	Priberam	0.0	UWr-VL	27.2	Priberam	0.0	CTC-NER	39.2
		L3i	0.0	L3i	0.0	L3i	0.0	Priberam	0.0	L3i	0.0	Priberam	0.0
		PAISC	0.0	PAISC	0.0	PAISC	0.0	L3i	0.0	PAISC	0.0	L3i	0.0
								PAISC	0.0			PAISC	0.0
Entity linking	Document	UWr-VL	37.6	TLD	47.0	UWr-VL	61.2	TLD	42.5	UWr-VL	52.0	TLD	48.9
	level	TLD	24.6	UWr-VL	46.0	TLD	44.7	UWr-VL	30.5	TLD	45.2	UWr-VL	45.3
		L3i	13.3	UL FRI	29.8	UL FRI	26.4	UL FRI	20.4	UL FRI	29.6	UL FRI	24.7
		Priberam	12.4	Priberam	23.9	PAISC	20.4	Priberam	15.5	Priberam	16.8	Priberam	23.7
		TraSpaS	11.5	L3i	22.5	L3i	20.3	PAISC	13.8	L3i	15.6	L3i	22.3
		PAISC	11.4	TraSpaS	22.1	Priberam	20.0	L3i	13.3	TraSpaS	14.9	TraSpaS	22.0
		UL FRI	6.1	PAISC	21.2	TraSpaS	18.4	TraSpaS	12.2	PAISC	13.8	PAISC	17.8
								CTC-NER	3.5			CTC-NER	2.3
	Single	UWr-VL	67.9	TLD	66.5	UWr-VL	73.0	TLD	47.4	UWr-VL	66.4	TLD	61.7
	language	TLD	57.1	UWr-VL	66.1	TLD	67.8	UWr-VL	38.9	TLD	59.2	UWr-VL	61.5
		PAISC	16.4	UL FRI	40.2	UL FRI	38.8	UL FRI	20.1	UL FRI	32.7	UL FRI	36.8
		L3i	10.9	PAISC	15.9	PAISC	13.7	Priberam	6.2	TraSpaS	10.0	Priberam	15.9
		Priberam	8.7	L3i	11.2	Priberam	9.3	L3i	4.2	Priberam	7.2	L3i	7.7
		UL FRI	7.6	TraSpaS	11.2	TraSpaS	8.2	PAISC	3.5	L3i	4.2	PAISC	7.5
		TraSpaS	3.6	Priberam	8.0	L3i	7.9	TraSpaS	2.0	PAISC	1.8	TraSpaS	6.3
								CTC-NER	1.8			CTC-NER	2.6

Table 4: F1-measure results for the COVID-19 corpus.

novation programme under grants 770299 (NewsEye).

Work on Slovene was financed through the European Union's Horizon 2020 Research and Innovation Programme under grant agreement No 825153, Project EMBEDDIA: Cross-Lingual Embeddings for Less-Represented Languages in European News Media, as well as Slovenian Research Agency's project: Computer-assisted multilingual news discourse analysis with contextual embeddings (CANDAS, J6-2581).

References

Željko Agić and Nikola Ljubešić. 2014. The SETimes.HR linguistically annotated corpus of Croatian. In *Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 1724–1727, Reykjavík, Iceland.

AY Antonova and AN Soloviev. 2013. Conditional random field models for the processing of Russian. In *Computational Linguistics and Intellectual Technologies: Papers From the Annual Conference "Dialogue"(Bekasovo, 29 May–2 June 2013)*, volume 1, pages 27–44.

Krešimir Baksa, Dino Golović, Goran Glavaš, and Jan Šnajder. 2017. Tagging named entities in Croatian tweets. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 4(1):20–41.

Luis Adrián Cabrera-Diego, Jose G. Moreno, and Antoine Doucet. 2021. Using a frustratingly easy domain and tagset adaptation for creating slavic named entity recognition systems. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. European Association for Computational Linguistics.

Nancy Chinchor. 1998. Overview of MUC-7/MET-2. In *Proceedings of Seventh Message Understanding Conference (MUC-7)*.

George R. Doddington, Alexis Mitchell, Mark A. Przybocki, Lance A. Ramshaw, Stephanie Strassel, and Ralph M. Weischedel. 2004. The Automatic Content Extraction (ACE) program—tasks, data, and evaluation. In *Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 837–840, Lisbon, Portugal.

Xiangyu Duan, Rafael E. Banchs, Min Zhang, Haizhou Li, and A. Kumaran. 2016. Report of NEWS 2016 machine transliteration shared task. In *Proceedings of The Sixth Named Entities Workshop*, pages 58–72, Berlin, Germany.

Pedro Ferreira, Ruben Cardoso, and Afonso Mendes. 2021. Priberam labs at the 3rd shared task on

USA 2020 ELECTIONS		*Language*											
Phase	*Metric*	bg		cs		pl		ru		sl		uk	
Recognition	Relaxed Partial	L3i	89.8	UWr-VL	91.3	Priberam	92.3	L3i	83.7	Priberam	91.5	TLD	84.6
		Priberam	88.7	Priberam	90.7	L3i	92.0	Priberam	83.4	L3i	91.5	Priberam	84.6
		TraSpaS	88.1	L3i	90.2	TLD	90.8	TraSpaS	81.5	UWr-VL	90.4	L3i	84.5
		UWr-VL	87.3	TLD	88.5	UWr-VL	89.8	TLD	80.9	TLD	89.8	TraSpaS	83.3
		TLD	87.3	UL FRI	88.4	TraSpaS	89.2	UL FRI	80.5	TraSpaS	89.4	UWr-VL	83.3
		UL FRI	86.9	TraSpaS	87.8	UL FRI	89.1	UWr-VL	77.2	UL FRI	88.6	UL FRI	83.2
		PAISC	83.6	PAISC	82.6	PAISC	66.4	PAISC	77.1	PAISC	86.0	PAISC	77.0
								CTC-NER	75.4			CTC-NER	71.1
Normalization		UWr-VL	51.3	UWr-VL	51.9	UWr-VL	62.1	TraSpaS	50.7	UWr-VL	62.4	UL FRI	56.9
		UL FRI	21.9	TraSpaS	42.0	TLD	51.0	UL FRI	48.8	UL FRI	43.9	TraSpaS	56.8
		TLD	19.1	TLD	40.1	UL FRI	50.1	TLD	46.5	TraSpaS	34.2	TLD	55.3
		TraSpaS	17.9	UL FRI	39.7	TraSpaS	42.4	CTC-NER	44.8	TLD	31.9	CTC-NER	36.9
		Priberam	0.0	Priberam	0.0	Priberam	0.0	UWr-VL	25.6	Priberam	0.0	UWr-VL	26.5
		L3i	0.0	L3i	0.0	L3i	0.0	Priberam	0.0	L3i	0.0	Priberam	0.0
		PAISC	0.0	PAISC	0.0	PAISC	0.0	L3i	0.0	PAISC	0.0	L3i	0.0
								PAISC	0.0			PAISC	0.0
Entity linking	Document level	UWr-VL	63.7	UWr-VL	64.3	UWr-VL	67.1	TLD	44.8	UWr-VL	67.3	UWr-VL	58.9
		TLD	58.7	TLD	55.3	TLD	62.3	UWr-VL	35.8	TLD	59.3	TLD	52.2
		Priberam	12.5	UL FRI	37.5	UL FRI	44.9	UL FRI	32.2	UL FRI	43.3	UL FRI	28.8
		L3i	12.1	L3i	30.5	Priberam	18.2	Priberam	12.3	L3i	18.3	Priberam	25.4
		TraSpaS	11.7	Priberam	29.5	L3i	18.0	L3i	12.3	Priberam	17.9	L3i	23.9
		PAISC	11.4	TraSpaS	28.6	TraSpaS	17.4	PAISC	9.9	TraSpaS	17.1	TraSpaS	23.5
		UL FRI	4.5	PAISC	21.6	PAISC	13.4	TraSpaS	9.8	PAISC	15.8	PAISC	16.8
								CTC-NER	2.8			CTC-NER	1.5
	Single language	UWr-VL	68.5	TLD	69.0	TLD	74.9	TLD	50.1	TLD	68.7	TLD	62.2
		TLD	67.1	UWr-VL	66.0	UWr-VL	69.9	UWr-VL	39.3	UWr-VL	66.5	UWr-VL	52.9
		PAISC	12.8	UL FRI	50.0	UL FRI	37.7	UL FRI	13.6	UL FRI	21.3	UL FRI	23.0
		Priberam	10.1	L3i	18.1	Priberam	14.8	Priberam	5.6	Priberam	8.4	TraSpaS	21.4
		TraSpaS	8.6	Priberam	17.7	L3i	14.5	L3i	5.5	L3i	8.3	L3i	20.5
		L3i	8.6	TraSpaS	17.7	TraSpaS	13.4	TraSpaS	5.1	TraSpaS	8.2	Priberam	20.2
		UL FRI	8.3	PAISC	14.1	PAISC	10.7	PAISC	4.4	PAISC	7.2	PAISC	12.9
								CTC-NER	3.6			CTC-NER	9.4

Table 5: Evaluation results (F1-measure) for the USA 2020 ELECTION corpus.

	COVID-19						USA 2020 ELECTIONS					
	bg	cs	pl	ru	sl	uk	bg	cs	pl	ru	sl	uk
Per	98.0	98.1	98.3	83.1	98.2	96.6	93.6	97.4	94.2	93.1	96.3	98.7
Loc	95.8	96.4	96.7	95.1	95.7	97.3	97.5	96.9	97.6	93.1	98.2	93.8
Org	86.5	89.4	91.3	82.9	88.8	87.6	86.6	89.6	86.3	76.6	76.7	81.7
Pro	55.1	76.2	75.6	47.6	63.4	49.4	80.7	87.4	90.2	66.9	77.7	69.9
Evt	52.6	40.1	57.8	52.6	63.5	75.9	29.6	26.1	40.5	55.7	38.0	16.1

Table 6: Recognition F1-measure (relaxed partial) by entity type—best-performing systems for each language.

slavner. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. European Association for Computational Linguistics.

Silja Huttunen, Roman Yangarber, and Ralph Grishman. 2002. Diversity of scenarios in information extraction. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002)*, Las Palmas, Spain.

Heng Ji, Joel Nothman, and Ben Hachey. 2014. Overview of TAC-KBP2014 entity discovery and linking tasks. In *Proceedings of Text Analysis Conference (TAC2014)*, pages 1333–1339.

Heng Ji, Joel Nothman, and Ben Hachey. 2015. Overview of TAC-KBP2015 tri-lingual entity discovery and linking. In *Proceedings of Text Analysis Conference (TAC2015)*.

Mladen Karan, Goran Glavaš, Frane Šarić, Jan Šnajder, Jure Mijić, Artur Šilić, and Bojana Dalbelo Bašić. 2013. CroNER: Recognizing named entities in Croatian using conditional random fields. *Informatica*, 37(2):165.

Michal Konkol and Miloslav Konopík. 2013. CRF-based Czech named entity recognizer and consolidation of Czech NER research. In *Text, Speech and Dialogue*, volume 8082 of *Lecture Notes in Computer Science*, pages 153–160. Springer Berlin Heidelberg.

A Kumaran, Mitesh M. Khapra, and Haizhou Li. 2010. Report of NEWS 2010 transliteration mining shared task. In *Proceedings of the 2010 Named Entities Workshop*, pages 21–28, Uppsala, Sweden.

Nikola Ljubešić, Marija Stupar, Tereza Jurić, and Željko Agić. 2013. Combining available datasets for building named entity recognition models of Croatian and Slovene. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 1(2):35–57.

Michał Marcińczuk, Marcin Oleksy, and Jan Kocoń. 2017. Inforex - a collaborative system for text cor-

pora annotation and analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, Varna, Bulgaria, September 2-8, 2017*, pages 473–482. INCOMA Ltd.

Nafise Sadat Moosavi and Michael Strube. 2016. Which coreference evaluation metric do you trust? A proposal for a link-based entity aware metric. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pages 632–642, Berlin, Germany.

Maciej Ogrodniczuk and Łukasz Kobyliński, editors. 2018. *Proceedings of the PolEval 2018 Workshop*. Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland.

Maciej Ogrodniczuk and Łukasz Kobyliński, editors. 2020. *Proceedings of the PolEval 2020 Workshop*. Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland.

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.

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*. Association for Computational Linguistics.

Jakub Piskorski, Karol Wieloch, and Marcin Sydow. 2009. On knowledge-poor methods for person name matching and lemmatization for highly inflectional languages. *Information retrieval*, 12(3):275–299.

Marko Prelevikj and Slavko Zitnik. 2021. Bsnlp 2021 shared task: Multilingual named entity recognition and matching using bert and dedupe for slavic languages. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. European Association for Computational Linguistics.

Adam Przepiórkowski. 2007. Slavonic information extraction and partial parsing. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing: Information Extraction and Enabling Technologies*, ACL '07, pages 1–10, Stroudsburg, PA, USA. Association for Computational Linguistics.

Paweł Rychlikowski, Adrian Lancucki, Adam Kaczmarek, Bartłomiej Najdecki, Adam Wawrzyński, and Wojciech Janowski. 2021. Named entity recognition and linking augmented with large-scale structured data. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. European Association for Computational Linguistics.

Agata Savary and Jakub Piskorski. 2011. Language Resources for Named Entity Annotation in the National Corpus of Polish. *Control and Cybernetics*, 40(2):361–391.

Magda Ševčíková, Zdeněk Žabokrtskỳ, and Oldřich Kruza. 2007. Named entities in Czech: annotating data and developing NE tagger. In *International Conference on Text, Speech and Dialogue*, pages 188–195. Springer.

Tadej Štajner, Tomaž Erjavec, and Simon Krek. 2013. Razpoznavanje imenskih entitet v slovenskem besedilu. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 1(2):58–81.

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 *Computational Linguistics and Intellectual Technologies. Proceedings of the Annual International Conference "Dialogue"*, pages 688–705.

Marek Suppa and Ondrej Jariabka. 2021. Benchmarking pre-trained language models for multilingual ner: Traspas at the bsnlp2021 shared task. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. European Association for Computational Linguistics.

Erik Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

Erik 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 - Volume 4*, CONLL '03, pages 142–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rinalds Vīksna and Inguna Skadina. 2021. Multilingual slavic named entity recognition. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. European Association for Computational Linguistics.

Jakub Waszczuk, Katarzyna Głowińska, Agata Savary, and Adam Przepiórkowski. 2010. Tools and methodologies for annotating syntax and named entities in the National Corpus of Polish. In *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT 2010): Computational Linguistics – Applications (CLA'10)*, pages 531–539, Wisła, Poland. PTI.

G Extending Neural Keyword Extraction with TF-IDF tagset matching

Extending Neural Keyword Extraction with TF-IDF tagset matching

Boshko Koloski
Jožef Stefan Institute
Jožef Stefan IPS
Jamova 39, Ljubljana
boshko.koloski@ijs.si

Senja Pollak
Jožef Stefan Institute
Jamova 39, Ljubljana
senja.pollak@ijs.si

Blaž Škrlj
Jožef Stefan Institute
Jožef Stefan IPS
Jamova 39, Ljubljana
blaz.skrlj@ijs.si

Matej Martinc
Jožef Stefan Institute
Jamova 39, Ljubljana
matej.martinc@ijs.si

Abstract

Keyword extraction is the task of identifying words (or multi-word expressions) that best describe a given document and serve in news portals to link articles of similar topics. In this work, we develop and evaluate our methods on four novel data sets covering less-represented, morphologically-rich languages in European news media industry (Croatian, Estonian, Latvian, and Russian). First, we perform evaluation of two supervised neural transformer-based methods, Transformer-based Neural Tagger for Keyword Identification (TNT-KID) and Bidirectional Encoder Representations from Transformers (BERT) with an additional Bidirectional Long Short-Term Memory Conditional Random Fields (BiLSTM CRF) classification head, and compare them to a baseline Term Frequency - Inverse Document Frequency (TF-IDF) based unsupervised approach. Next, we show that by combining the keywords retrieved by both neural transformer-based methods and extending the final set of keywords with an unsupervised TF-IDF based technique, we can drastically improve the recall of the system, making it appropriate for usage as a recommendation system in the media house environment.

1 Introduction

Keywords are words (or multi-word expressions) that best describe the subject of a document, effectively summarise it and can also be used in several document categorization tasks. In online news portals, keywords help with efficient retrieval of articles when needed. Similar keywords characterise articles of similar topics, which can help editors to link related articles, journalists to find similar articles and readers to retrieve articles of interest when browsing the portals. For journalists manually assigning tags (keywords) to articles represents a demanding task, and high-quality automated keyword extraction shows to be one of components in news digitalization process that many media houses seek for.

The task of keyword extraction can generally be tackled in an unsupervised way, i.e., by relying on frequency based statistical measures (Campos et al., 2020) or graph statistics (Škrlj et al., 2019), or with a supervised keyword extraction tool, which requires a training set of sufficient size and from appropriate domain. While supervised methods tend to work better due to their ability to adapt to a specifics of the syntax, semantics, content, genre and keyword assignment regime of a specific text (Martinc et al., 2020a), their training for some less resource languages is problematic due to scarcity of large manually annotated resources. For this reason, studies about supervised keyword extraction conducted on less resourced languages are still very rare. To overcome this research gap, in this paper we focus on supervised keyword extraction on three less resourced languages, Croatian, Latvian, and Estonian, and one fairly well resourced language (Russian) and conduct experiments on data sets of media partners in the EMBEDDIA project[1]. The code for the experiments is made available on GitHub under the MIT license[2].

In media house environments, automatic keyword extraction systems are expected to return a diverse list of keyword candidates (of constant length), which is then inspected by a journalist who

[1] http://embeddia.eu/

[2] https://github.com/bkolosk1/Extending-Neural-Keyword-Extraction-with-TF-IDF-tagset-matching/

Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation, pages 22–29

manually selects appropriate candidates. While the state-of-the-art supervised approaches in most cases offer good enough precision for this type of usage as a recommendation system, the recall of these systems is nevertheless problematic. Supervised systems learn how many keywords should be returned for each news article on the gold standard train set, which generally contains only a small amount of manually approved candidates for each news article. For example, among the datasets used in our experiments (see Section 3), the Russian train set contains the most (on average 4.44) present keywords (i.e., keywords which appear in the text of the article and can be used for training of the supervised models) per article, while the Croatian test set contains only 1.19 keywords per article. This means that for Croatian, the model will learn to return around 1.19 keywords for each article, which is not enough.

To solve this problem we show that we can improve the recall of the existing supervised keyword extraction system by:

- Proposing an additional TF-IDF tagset matching technique, which finds additional keyword candidates by ranking the words in the news article that have appeared in the predefined keyword set containing words from the gold standard train set. The new hybrid system first checks how many keywords were returned by the supervised approach and if the number is smaller than needed, the list is expanded by the best ranked keywords returned by the TF-IDF based extraction system.

- Combining the outputs of several state-of-the-art supervised keyword extraction approaches.

The rest of this work is structured as follows: Section 2 presents the related work, while Section 3 describes the datasets on which we evaluate our method. Section 4 describes our proposed method with all corresponding steps. The experiment settings are described in Section 5 and the evaluation of the proposed methods is shown in Section 6. The conclusions and the proposed further work are presented in Section 7.

2 Related Work

Many different approaches have been developed to tackle the problem of extracting keywords. The early approaches, such as KP-MINER (El-Beltagy and Rafea, 2009) and RAKE (Rose et al., 2010) rely on unsupervised techniques which employ frequency based metrics for extraction of keywords from text. Formally, aforementioned approaches search for the words w from vocabulary $\mathcal{V}$ that maximize a given metric h for a given text t:

$$\text{kw} = \underset{w \in \mathcal{V}}{\operatorname{argmax}}\, h(w, t).$$

In these approaches, frequency is of high relevance and it is assumed that the more frequent a given word, the more important the meaning this word carries for a given document. Most popular such metrics are the naïve frequency (word count) and the term frequency-inverse document frequency (TF-IDF) (Salton and McGill, 1986).

Most recent state-of-the-art statistical approaches, such as YAKE (Campos et al., 2020), also employ frequency based features, but combine them with other features such as casing, position, relatedness to context and dispersion of a specific term in order to derive a final score for each keyword candidate.

Another line of research models this problem by exploiting concepts from graph theory. Approaches, such as TextRank (Mihalcea and Tarau, 2004), Single Rank (Wan and Xiao, 2008), TopicRank (Bougouin et al., 2013) and Topical PageRank (Sterckx et al., 2015) build a graph G, i.e., a mathematical construct described by a set of vertexes V and a set of edges E connecting two vertices. In one of the most recent approaches called RaKUn (Škrlj et al., 2019), a directed graph is constructed from text, where vertexes V and two words w_i, w_{i+1} are linked if they appear following one another. Keywords are ranked by a shortest path-based metric from graph theory - the load centrality.

The task of keyword extraction can also be tackled in a supervised way. One of the first supervised approaches was an algorithm named KEA (Witten et al., 2005), which uses only TF-IDF and the term's position in the text as features for term identification. More recent neural approaches to keyword detection consider the problem as a sequence-to-sequence generation task (Meng et al., 2017) and employ a generative model for keyword prediction with a recurrent encoder-decoder framework and an attention mechanism capable of detecting keywords in the input text sequence whilst also potentially finding keywords that do not appear in the text.

Finally, the newest branch of models consider keyword extraction as a sequence labelling task and tackle keyword detection with transformers. Sahrawat et al. (2020) fed contextual embeddings generated by several transformer models (BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), GPT-2 (Radford et al., 2019), etc.) into two types of neural architectures, a bidirectional Long short-term memory network (BiLSTM) and a BiLSTM network with an additional Conditional random fields layer (BiLSTM-CRF). Sun et al. (2020) on the other hand proposed BERT-JointKPE that employs a chunking network to identify phrases and a ranking network to learn their salience in the document. By training BERT jointly on the chunking and ranking tasks the model manages to establish balance between the estimation of keyphrase quality and salience.

Another state-of-the-art transformer based approach is TNT-KID (Transformer-based Neural Tagger for Keyword Identification) (Martinc et al., 2020a), which does not rely on pretrained language models such as BERT, but rather allows the user to train their own language model on the appropriate domain. The study shows that smaller unlabelled domain specific corpora can be successfully used for unsupervised pretraining, which makes the proposed approach easily transferable to low-resource languages. It also proposes several modifications to the transformer architecture in order to adapt it for a keyword extraction task and improve performance of the model.

3 Data Description

We conducted experiments on datasets containing news in four languages; Latvian, Estonian, Russian, and Croatian. Latvian, Estonian and Russian datasets contain news from the Ekspress Group, specifically from Estonian Ekspress Meedia (news in Estonian and Russian) and from Latvian Delfi (news in Latvian and Russian). The dataset statistics are presented in Table 2, and the datasets (Pollak et al., 2021) and their train/test splits[3] are publicly available. The media-houses provided news articles from 2015 up to the 2019. We divided them into training and test sets. For the Latvian, Estonian, and Russian training sets, we used the articles from 2018, while for the test set the articles from 2019 were used. For Croatian, the articles from 2019 are arranged by date and split into training and test (i.e., about 10% of the 2019 articles with the most recent date) set. In our study, we also use tagsets of keywords. Tagset corresponds either to a collection of keywords maintained by editors of a media house (see e.g. Estonian tagset), or to a tagset constructed from assigned keywords from articles available in the training set. The type of tagset and the number of unique tags for each language are listed in Table 1.

Dataset	Unique tags	Type of tags
Croatian	21,165	Constructed
Estonian	52,068	Provided
Russian	5,899	Provided
Latvian	4,015	Constructed

Table 1: Distribution of tags provided per language. The media houses provided tagsets for Estonian and Russian, while the tags for Latvian and Croatian were extracted from the train set.

4 Methodology

The recent supervised neural methods are very precise, but, as was already mentioned in Section 1, in same cases they do not return a sufficient number of keywords. This is due to the fact that the methods are trained on the training data with a low number of gold standard keywords (as it can be seen from Table 2). To meet the media partners' needs, we designed a method that complements state-of-the-art neural methods (the TNT-KID method (Martinc et al., 2020b) and the transformer-based method proposed by Sahrawat et al. (2020), which are both described in Section 2) by a tagset matching approach, returning constant number of keywords (k=10).

4.1 Transformer-based Keyword Extraction

Both supervised neural approaches employed in this study are based on the Transformer architecture (Vaswani et al., 2017), which was somewhat adapted for the specific task at hand. Both models are fed lowercased text consisting of the title and the body of the article. Tokenization is conducted by either using the default BERT tokenizer (when BERT is used) or by employing Sentencepiece tokenizer (Kudo and Richardson, 2018) (when TNT-KID is used). While the multilingual BERT model is already pretrained on a large corpus consisting of Wikipedias of about 100 languages (Devlin et al.,

[3] https://www.clarin.si/repository/xmlui/handle/11356/1403

			Avg. Train					Avg. Test				
Dataset	**Total docs**	**Total kw.**	**Total docs**	**Doc len**	**Kw.**	**% present kw.**	**present kw.**	**Total docs**	**Doc len**	**Kw.**	**% present kw.**	**Present kw.**
Croatian	35,805	126,684	32,223	438.50	3.54	0.32	1.19	3582	464.39	3.53	0.34	1.26
Estonian	18,497	59,242	10,750	395.24	3.81	0.65	2.77	7,747	411.59	4.09	0.69	3.12
Russian	25,306	5,953	13,831	392.82	5.66	0.76	4.44	11,475	335.93	5.43	0.79	4.33
Latvian	24,774	4,036	13,133	378.03	3.23	0.53	1.69	11,641	460.15	3.19	0.55	1.71

Table 2: Media partners' datasets used for empirical evaluation of keyword extraction algorithms.

2018), TNT-KID requires an additional language model pretraining on the domain specific corpus.

4.2 TF-IDF(tm) Tagset Matching

In our approach, we first take the keywords returned by a neural keyword extraction method and next complement the returned keyword list by adding the missing keywords to achieve the set goal of k keywords. The added keywords are selected by taking the top-ranked candidates from the TF-IDF tagset matching extraction conducted on the preprocessed news articles and keywords.

4.2.1 Preprocessing

First, we concatenate the body and the title of the article. After that we lowercase the text and remove stopwords. Finally, the text is tokenized and lemmatized with the Lemmagen3 lemmatizer (Juršič et al., 2010), which supports lemmatization for all the languages except Latvian. For Latvian we use the LatvianStemmer [4]. For the stopword removal we used the *Stopwords-ISO* [5] Python library which contained stopwords for all four languages. The final cleaned textual input consists of the concatenation of all of the preprocessed words from the document. We apply the same preprocessing procedure on the predetermined tagsets for each language. The preprocessing procedure is visualized in Figure 1.

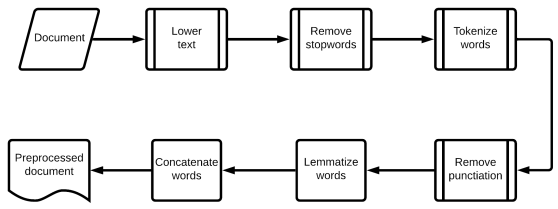


Figure 1: Preprocessing pipeline used for the document normalization and cleaning.

[4] https://github.com/rihardsk/LatvianStemmer

[5] https://github.com/stopwords-iso

4.2.2 TF-IDF Weighting Scheme

The TF-IDF weighting scheme (Salton and McGill, 1986) assigns each word its weight w based on the frequency of the word in the document (term frequency) and the number of documents the word appears in (inverse document frequency). More specifically, TF-IDF is calculated with the following equation:

$$TF - IDF_{\mathrm{i}} = tf_{\mathrm{i,j}} \cdot \log_e(\frac{|D|}{df_i})$$

The formula has two main components:

- *Term-frequency* (tf) that counts the number of appearances of a word in the document (in the equation above, $tf_{\mathrm{i,j}}$ denotes the number of occurrences of the word i in the document j)
- *Inverse-document-frequency* (idf) ensures that words appearing in more documents are assigned lower weights (in the formula above df_i is the number of documents containing word i and $|D|$ denotes the number of documents).

The assumption is that words with a higher TF-IDF value are more likely to be keywords.

4.3 Tagset Matching Keyword Expansion

For a given neural keyword extraction method *N*, and for each document *d*, we select l best ranked keywords according to the TF-IDF(tm), which appear in the keyword tagset for each specific dataset. Here, l corresponds to k - m, where $k = 10$ and m corresponds to the number of keywords returned by a neural method.

Since some of the keywords in the tagsets provided by the media partners were variations of the same root word (i.e., keywords are not lemmatized), we created a mapping from a root word (i.e., a word lemma or a stem) to a list of possible variations in the keyword dataset. For example, a word *'riigieksam'* (*'exam'*) appearing in the article, could be mapped to three tags in the tagset by the Estonian media house with the same root form *'riigieksam'*: *'riigieksamid'*, *'riigieksamide'* and *'riigieksam'*.

We tested several strategies for mapping the occurrence of a word in the news article to a specific tag in the tagset. For each lemma that mapped to multiple tags, we tested returning a random tag, a tag with minimal length and a tag of maximal length. In the final version, we opted to return the tag with the minimal length, since this tag corresponded to the lemma of the word most often.

5 Experimental Settings

We conducted experiments on the datasets described in Section 3. We evaluate the following methods and combinations of methods:

- **TF-IDF(tm):** Here, we employ the preprocessing and TF-IDF-based weighting of keywords described in Section 4 and select the top-ranked keywords that are present in the tagset.
- **TNT-KID** (Martinc et al., 2020b): For each dataset, we first pretrain the model with an autoregressive language model objective. After that, the model is fine-tuned on the same train set for the keyword extraction task. Sequence length was set to 256, embedding size to 512 and batch size to 8, and we employ the same preprocessing as in the original study (Martinc et al., 2020b).
- **BERT + BiLSTM-CRF** (Sahrawat et al., 2020): We employ an uncased multilingual BERT[6] model with an embedding size of 768 and 12 attention heads, with an additional BiLSTM-CRF token classification head, same as in Sahrawat et al. (2020).
- **TNT-KID & BERT + BiLSTM-CRF**: We extracted keywords with both of the methods and complemented the TNT-KID extracted keywords with the BERT + BiLSTM-CRF extracted keywords in order to retrieve more keywords. Duplicates (i.e., keywords extracted by both methods) are removed.
- **TNT-KID & TF-IDF**: If the keyword set extracted by TNT-KID contains less than 10 keywords, it is expanded with keywords retrieved with the proposed TF-IDF(tm) approach, i.e., best ranked keywords according to TF-IDF, which do not appear in the keyword set extracted by TNT-KID.
- **BERT + BiLSTM-CRF & TF-IDF**: If the keyword set extracted by BERT + BiLSTM-CRF contains less than 10 keywords, it is expanded with keywords retrieved with the proposed TF-IDF(tm) approach, i.e., best ranked keywords according to TF-IDF, which do not appear in the keyword set extracted by BERT + BiLSTM-CRF.
- **TNT-KID & BERT + BiLSTM-CRF & TF-IDF:** the keyword set extracted with the TNT-KID is complemented by keywords extracted with BERT + BiLSTM-CRF (duplicates are removed). If after the expansion the keyword set still contains less than 10 keywords, it is expanded again, this time with keywords retrieved by the TF-IDF(tm) approach.

For TNT-KID, which is the only model that requires language model pretraining, language models were trained on train sets in Table 2 for up to ten epochs. Next, TNT-KID and BERT + BiLSTM-CRF were fine-tuned on the training datasets, which were randomly split into 80 percent of documents used for training and 20 percent of documents used for validation. The documents containing more than 256 tokens are truncated, while the documents containing less than 256 tokens are padded with a special $<$ pad $>$ token at the end. We fine-tuned each model for a maximum of 10 epochs and after each epoch the trained model was tested on the documents chosen for validation. The model that showed the best performance on this set of validation documents (in terms of F@10 score) was used for keyword detection on the test set.

6 Evaluation

For evaluation, we employ precision, recall and F1 score. While F1@10 and recall@10 are the most relevant metrics for the media partners, we also report precision@10, precision@5, recall@5 and F1@5. Only keywords which appear in a text (present keywords) were used as a gold standard, since we only evaluate approaches for keyword tagging that are not capable of finding keywords which do not appear in the text. Lowercasing and lemmatization (stemming in the case of Latvian) are performed on both the gold standard and the

[6]More specifically, we use the 'bert-base-multilingual-uncased' implementation of BERT from the Transformers library (https://github.com/huggingface/transformers).

Model	P@5	R@5	F1@5	P@10	R@10	F1@10
Croatian						
TF-IDF	0.2226	0.4543	0.2988	0.1466	0.5888	0.2347
TNT-KID	0.3296	0.5135	0.4015	0.3167	0.5359	0.3981
BERT + BiLSTM-CRF	**0.4607**	0.4672	**0.4640**	**0.4599**	0.4708	**0.4654**
TNT-KID & TF-IDF(tm)	0.2659	0.5670	0.3621	0.1688	0.6944	0.2716
BERT + BiLSTM-CRF & TF-IDF(tm)	0.2644	0.5656	0.3604	0.1549	0.6410	0.2495
TNT-KID & BERT + BiLSTM-CRF	0.2940	0.5447	0.3820	0.2659	0.5968	0.3679
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.2648	**0.5681**	0.3612	0.1699	**0.7040**	0.2738
Estonian						
TF-IDF	0.0716	0.1488	0.0966	0.0496	0.1950	0.0790
TNT-KID	**0.5194**	0.5676	**0.5424**	**0.5098**	0.5942	**0.5942**
BERT + BiLSTM-CRF	0.5118	0.4617	0.4855	0.5078	0.4775	0.4922
TNT-KID & TF-IDF(tm)	0.3463	0.5997	0.4391	0.1978	0.6541	0.3037
BERT + BiLSTM-CRF & TF-IDF(tm)	0.3175	0.4978	0.3877	0.1789	0.5381	0.2686
TNT-KID & BERT + BiLSTM-CRF	0.4421	0.6014	0.5096	0.4028	0.6438	0.4956
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.3588	**0.6206**	0.4547	0.2107	**0.6912**	0.3230
Russian						
TF-IDF	0.1764	0.2314	0.2002	0.1663	0.3350	0.2223
TNT-KID	**0.7108**	0.6007	**0.6512**	**0.7038**	0.6250	**0.6621**
BERT + BiLSTM-CRF	0.6901	0.5467	0.5467	0.6849	0.5643	0.6187
TNT-KID & TF-IDF(tm)	0.4519	0.6293	0.5261	0.2981	0.6946	0.4172
BERT + BiLSTM-CRF & TF-IDF(tm)	0.4157	0.5728	0.4818	0.2753	0.6378	0.3846
TNT-KID & BERT + BiLSTM-CRF	0.6226	0.6375	0.6300	0.5877	0.6707	0.6265
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.4622	**0.6527**	0.5412	0.2965	**0.7213**	0.4203
Latvian						
TF-IDF	0.2258	0.5035	0.3118	0.1708	0.5965	0.2655
TNT-KID	0.6089	0.6887	**0.6464**	0.6054	0.6960	**0.6476**
BERT + BiLSTM-CRF	**0.6215**	0.6214	0.6214	**0.6204**	0.6243	0.6223
TNT-KID & TF-IDF(tm)	0.3402	**0.7934**	0.4762	0.2253	0.8653	0.3575
BERT + BiLSTM-CRF & TF-IDF(tm)	0.2985	0.6957	0.4178	0.1889	0.7427	0.3012
TNT-KID & BERT + BiLSTM-CRF	0.4545	0.7189	0.5569	0.4341	0.7297	0.5443
TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm)	0.3318	0.7852	0.4666	0.2124	**0.8672**	0.3414

Table 3: Results on the EMBEDDIA media partner datasets.

extracted keywords (keyphrases) during the evaluation. The results of the evaluation on all four languages are listed in Table 3.

Results suggest, that neural approaches, TNT-KID and BERT+BiLSTM-CRF offer comparable performance on all datasets but nevertheless achieve different results for different languages. TNT-KID outperforms BERT-BiLSTM-CRF model according to all the evaluation metrics on the Estonian and Russian news dataset. It also outperforms all other methods in terms of precision and F1 score. On the other hand, BERT+BiLSTM-CRF performs better on the Croatian dataset in terms of precision and F1-score. On Latvian TNT-KID achieves top results in terms of F1, while BERT+BiLSTM-CRF offers better precision.

Even though the TF-IDF tagset matching method performs poorly on its own, we can nevertheless drastically improve the recall@5 and the recall@10 of both neural systems, if we expand the keyword tag sets returned by the neural methods with the TF-IDF ranked keywords. The improvement is substantial and consistent for all datasets, but it nevertheless comes at the expanse of the lower precision and F1 score. This is not surprising, since the final expanded keyword set always returns 10 keywords, i.e., much more than the average number of present gold standard keywords in the media partner datasets (see Table 2), which badly affects the precision of the approach. Nevertheless, since for a journalist a manual inspection of 10 keyword candidates per article and manual selection of good candidates (e.g., by clicking on them) still requires less time than the manual selection of keywords from an article, we argue that the improvement of recall at the expanse of the precision is a good trade

off, if the system is intended to be used as a recommendation system in the media house environment.

Combining keywords returned by TNT-KID and BERT + BiLSTM-CRF also consistently improves recall, but again at the expanse of lower precision and F1 score. Overall, for all four languages, the best performing method in terms of recall is the TNT-KID & BERT + BiLSTM-CRF & TF-IDF(tm).

7 Conclusion and Future Work

In this work, we tested two state-of-the-art neural approaches for keyword extraction, TNT-KID (Martinc et al., 2020a) and BERT BiLSTM-CRF (Sahrawat et al., 2020), on three less resourced European languages, Estonian, Latvian, Croatian, as well as on Russian. We also proposed a tagset based keyword expansion approach, which drastically improves the recall of the method, making it more suitable for the application in the media house environment.

Our study is one of the very few studies where supervised keyword extraction models were employed on several less resourced languages. The results suggest that these models perform well on languages other than English and could also be successfully leveraged for keyword extraction on morphologically rich languages.

The focus of the study was whether we can improve the recall of the supervised models, in order to make them more useful as recommendation systems in the media house environment. Our method manages to increase the number of retrieved keywords, which drastically improves the recall for all languages. For example, by combing all neural methods and the TF-IDF based approach, we improve on the recall@10 achieved by the best performing neural model, TNT-KID, by 16.81 percentage points for Croatian, 9.70 percentage points for Estonian, 9.63 percentage points for Russian and 17.12 percentage points for Latvian. The resulting method nevertheless offers lower precision, which we will try to improve in the future work.

In the future we also plan to perform a qualitative evaluation of our methods by journalists from the media houses. Next, we plan to explore how adding background knowledge from knowledge databases - lexical (e.g. Wordnet(Fellbaum, 1998)) or factual (e.g. WikiData(Vrandečić and Krötzsch, 2014)) would benefit the aforementioned methods. The assumption is that with the linkage of the text representation and the background knowledge we would achieve a more representative understanding of the articles and the concepts appearing in them, which would result in a more successful keyword extraction.

In traditional machine-learning setting a common practice of combining different classifier outputs to a single output is referred to as stacking. We propose further research on this topic by testing combinations of various keyword extraction models. Finally, we also plan to further improve our unsupervised TF-IDF based keyword extraction method. One way to to do this would be to add the notion of positional encoding, since some of the keywords in the news-media domain often can be found at the beginning of the article and the TF-IDF(tm) does not take this into account while applying the weighting on the matched terms.

8 Acknowledgements

This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). The third author was financed via young research ARRS grant. Finally, the authors acknowledge the financial support from the Slovenian Research Agency for research core funding for the programme Knowledge Technologies (No. P2-0103) and the project TermFrame - Terminology and Knowledge Frames across Languages (No. J6-9372).

References

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. Topicrank: Graph-based topic ranking for keyphrase extraction. In *International joint conference on natural language processing (IJCNLP)*, pages 543–551.

Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 509:257 – 289.

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*.

Samhaa R. El-Beltagy and Ahmed Rafea. 2009. Kp-miner: A keyphrase extraction system for english and arabic documents. *Inf. Syst.*, 34(1):132–144.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Matjaž Juršič, Igor Mozetič, Tomaž Erjavec, and Nada Lavrač. 2010. Lemmagen: Multilingual lemmatisation with induced ripple-down rules. *Journal of Universal Computer Science*, 16(9):1190–1214.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Matej Martinc, Blaž Škrlj, and Senja Pollak. 2020a. Tnt-kid: Transformer-based neural tagger for keyword identification. *arXiv preprint arXiv:2003.09166*.

Matej Martinc, Blaž Škrlj, and Senja Pollak. 2020b. Tnt-kid: Transformer-based neural tagger for keyword identification.

Rui Meng, Sanqiang Zhao, Shuguang Han, Daqing He, Peter Brusilovsky, and Yu Chi. 2017. Deep keyphrase generation. *arXiv preprint arXiv:1704.06879*.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Senja Pollak, Marko Robnik Šikonja, Matthew Purver, Michele Boggia, Ravi Shekhar, Marko Pranjić, Salla Salmela, Ivar Krustok, Tarmo Paju, Carl-Gustav Linden, Leo Leppänen, Elaine Zosa, Matej Ulčar, Linda Freienthal, Silver Traat, Luis Adrián Cabrera-Diego, Matej Martinc, Nada Lavrač, Blaž Škrlj, Martin Žnidaršič, Andraž Pelicon, Boshko Koloski, Vid Podpečan, Janez Kranjc, Shane Sheehan, Hannu Toivonen, Emanuela Boros, Jose Moreno, and Antoine Doucet. 2021. EMBEDDIA tools, datasets and challenges: Resources and hackathon contributions. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.

Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. Automatic keyword extraction from individual documents. *Text mining: applications and theory*, 1:1–20.

Dhruva Sahrawat, Debanjan Mahata, Mayank Kulkarni, Haimin Zhang, Rakesh Gosangi, Amanda Stent, Agniv Sharma, Yaman Kumar, Rajiv Ratn Shah, and Roger Zimmermann. 2020. Keyphrase extraction from scholarly articles as sequence labeling using contextualized embeddings. In *Proceedings of European Conference on Information Retrieval (ECIR 2020)*, pages 328–335.

Gerard Salton and Michael J McGill. 1986. Introduction to modern information retrieval.

Blaž Škrlj, Andraž Repar, and Senja Pollak. 2019. Rakun: Rank-based keyword extraction via unsupervised learning and meta vertex aggregation. In *International Conference on Statistical Language and Speech Processing*, pages 311–323. Springer.

Lucas Sterckx, Thomas Demeester, Johannes Deleu, and Chris Develder. 2015. Topical word importance for fast keyphrase extraction. In *Proceedings of the 24th International Conference on World Wide Web*, pages 121–122.

Si Sun, Chenyan Xiong, Zhenghao Liu, Zhiyuan Liu, and Jie Bao. 2020. Joint keyphrase chunking and salience ranking with bert. *arXiv preprint arXiv:2004.13639*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Xiaojun Wan and Jianguo Xiao. 2008. Single document keyphrase extraction using neighborhood knowledge. In *AAAI*, volume 8, pages 855–860.

Ian H Witten, Gordon W Paynter, Eibe Frank, Carl Gutwin, and Craig G Nevill-Manning. 2005. Kea: Practical automated keyphrase extraction. In *Design and Usability of Digital Libraries: Case Studies in the Asia Pacific*, pages 129–152. IGI global.

H Aligning Estonian and Russian news industry keywords with the help of subtitle translations and an environmental thesaurus

Aligning Estonian and Russian news industry keywords with the help of subtitle translations and an environmental thesaurus

Andraž Repar
International Postgraduate School / Jamova 39, 1000 Ljubljana, Slovenia
andraz.repar@ijs.si
Andrej Shumakov
Ekspress Meedia / Narva mnt 13, 10151 Tallinn, Estonia

Abstract

This paper presents the implementation of a bilingual term alignment approach developed by Repar et al. (2019) to a dataset of unaligned Estonian and Russian keywords which were manually assigned by journalists to describe the article topic. We started by separating the dataset into Estonian and Russian tags based on whether they are written in the Latin or Cyrillic script. Then we selected the available language-specific resources necessary for the alignment system to work. Despite the domains of the language-specific resources (subtitles and environment) not matching the domain of the dataset (news articles), we were able to achieve respectable results with manual evaluation indicating that almost 3/4 of the aligned keyword pairs are at least partial matches.

1 Introduction and related work

The ability to accurately align concepts between languages can provide significant benefits in many practical applications. For example, in terminology, terms can be aligned between languages to provide bilingual terminological resources, while in the news industry, keywords can be aligned to provide better news clustering or search in another language. Accurate bilingual resources can also serve as seed data for various other NLP tasks, such as multilingual vector space alignment.

In this paper, we describe the experiments on an Estonian-Russian dataset of news tags — labels that were manually assigned to news articles by journalists and editors at Ekspress Meedia, one of the largest news publishers in the Baltic region. The dataset contains both Estonian and Russian tags, but they are not aligned between the two languages. We adapted the machine learning term alignment approach described by Repar et al. (2019) to align the Russian and Estonian tags in the dataset.

The alignment approach in Repar et al. (2019) is a reproduction and adaptation of the approach described by Aker et al. (2013a). Repar et al. (2019) managed to reach a precision of over 0.9 and therefore approach the values presented by Aker et al. (2013a) by tweaking several parameters and developing new machine learning features. They also developed a novel cognate-based approach which could be effective in texts with a high proportion of novel terminology that cannot be detected by relying on dictionary-based features. In this work, we perform the implementation of the proposed method on a novel, Estonian-Russian language pair, and in a novel application of tagset alignment.

Section 1 lists the related work, Section 2 contains a description of the tag dataset used, Section 3 describes the system architecture, Section 4 explains the resources used in this paper, Section 5 contains the results of the experiments and Section 6 provides conclusions and future work.

2 Dataset description

The dataset of Estonian and Russian tags was provided by Ekspress Meedia as a simple list of one tag per line. The total number of tags was 65,830. The tagset consists of keywords that journalists assigne to articles to describe an article's topic, and was cut down recently by the editors from more than 210,000 tags.

The number of Russian tags was 6,198 and they were mixed with the Estonian tags in random order. Since Russian and Estonian use different writing scripts (Cyrillic vs Latin), we were able to separate the tags using a simple regular expression to detect Cyrillic characters. The vast majority of the tags are either unigrams or bigrams (see Table 1 for details).

Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation, pages 71–75

Grams	Estonian	Russian
1	0.49	0.49
2	0.44	0.41
3	0.05	0.06
4	0.01	0.02
> 4	0.01	0.02

Table 1: An analysis of the provided dataset in terms of multi-word units. The values represent the ratio of the total number of tags for the respective language. The total number of Estonian tags was 59,632, and the total number of Russian tags was 6,198. The largest Estonian tag was a 14-gram and the largest Russian tag was an 11-gram, but the vast majority of tags are either unigrams or bigrams.

3 System architecture

The algoritm used in this paper is based on the approach described in Repar et al. (2019) which is itself a replication and an adaptation of Aker et al. (2013b). The original approach designed by (Aker et al., 2013b) was developed to align terminology from comparable (or parallel) corpora using machine-learning techniques. They use terms from the Eurovoc (Steinberger et al., 2002) thesaurus and train an SVM binary classifier (Joachims, 2002) (with a linear kernel and the trade-off between training error and margin parameter c = 10). The task of bilingual alignment is treated as a binary classification - each term from the source language *S* is paired with each term from the target language *T* and the classifier then decides whether the aligned pair is correct or incorrect. (Aker et al., 2013b) use two types of features that express correspondences between the words (composing a term) in the target and source language:

- 7 dictionary-based (using Giza++) features which take advantage of dictionaries created from large parallel corpora of which 6 are direction-dependent (source-to-target or target-to-source) and 1 direction-independent - resulting in altogether 13 features, and
- 5 cognate-based (on the basis of (Gaizauskas et al., 2012)) which utilize string-based word similarity between languages.

To match words with morphological differences, they do not perform direct string matching but utilize Levenshtein Distance. Two words were considered equal if the Levenshtein Distance (Levenshtein, 1966) was equal or higher than 0.95. For closed-compounding languages, they check whether the compound source term has an initial prefix that matches the translation of the first target word, provided that translation is at least 5 characters long.

Additional features are also constructed by:

- Using language pair specific transliteration rules to create additional cognate-based features. The purpose of this task was to try to match the cognate terms while taking into account the differences in writing systems between two languages: e.g. Greek and English. Transliteration rules were created for both directions (source-to-target and target-to-source) separately and cognate-based features were constructed for both directions - resulting in additional 10 cognate-based features with transliteration rules.
- Combining the dictionary and cognate-based features in a set of combined features where the term pair alignment is correct if either the dictionary or the cognate-based method returns a positive result. This process resulted in additional 10 combined features[1].

A subset of the features is described below (For a full list of features, see Repar et al. (2019)):

- *isFirstWordTranslated*: A dictionary feature that checks whether the first word of the source term is a translation of the first word in the target term (based on the Giza++ dictionary).
- *longestTranslatedUnitInPercentage*: A dictionary feature representing the ratio of the longest contiguous sequence of source words which has a translation in the target term (compared to the source term length).
- *Longest Common Subsequence Ratio*: A cognate feature measuring the longest common non-consecutive sequence of characters between two strings
- *isFirstWordCovered*: A combined feature indicating whether the first word in the source

[1]For combined features, a word is considered as covered if it can be found in the corresponding set of Giza++ translations or if one of the cognate-based measures (Longest Common Subsequence, Longest Common Substring, Levensthein Distance, Needleman-Wunsch Distance, Dice) is 0.70 or higher (set experimentally by (Aker et al., 2013b))

term has a translation or transliteration in the target term.

- *isFirstWordCognate*: a binary feature which returns True if the longest common consecutive string (LCST) of the first words in the source and target terms divided by the length of the longest of the two words is greater than or equal to a threshold value of 0.7 and both words are longer than 3 characters.

Repar et al. (2019) start by reproducing this approach, but were unable to replicate the results. During the subsequent investigation, they discovered that using the same balance ratio in the training and test sets (i.e. 1:200, which was set by Aker et al. (2013b) to mimic real-world scenarios) have a significant impact on the performance of the algorithm. Furthermore, they filter training set term pairs based on term length and feature values (hence the different training set sizes in Table 2) and develop new cognate-based features.

The system requires several language-specific resources:

- A large parallel corpus to calculate word alignment probability with Giza++. The system in Repar et al. (2019) uses the DGT translation memory (Steinberger et al., 2013).
- A list of aligned terms that serve as training data. The system in Repar et al. (2019) uses the Eurovoc thesaurus (Steinberger et al., 2002). 600 Eurovoc term pairs are used as test data, while the rest is used for training.
- Transliteration rules for the construction of reverse cognate-based features (cognate features are constructed twice: first the target word is transliterated into the source language script, then the source word is transliterated in the target language script).

The constructed features are then used to train the SVM classifier which can be used to predict the alignment of terms between two languages.

4 Resources for the Estonian-Russian experiment

While the DGT translation memory and the Eurovoc thesaurus support all official EU languages, there is no Russian support since Russia is not an EU member state. In order to train the classifier, we therefore had to find alternative resources.

For the parallel corpus, we made experiments with the Estonian Open Parallel corpus[2] and the Estonian-Russian OpenSubtitles corpus from the Opus portal[3]. The OpenSubtitles corpus performed better, most likely due to its much larger size (85,449 parallel Estonian-Russian segments in the Estonian Open Parallel corpus vs. 7.1 million segments in the OpenSubtitles corpus).

While finding parallel Estonian-Russian corpora was trivial due the the list of available corpora on the Opus portal, finding an appropriate bilingual terminological database proved to be more difficult. Ideally, we would want to use a media or news-related Estonian-Russian terminological resource, but to the best of our knowledge, there was none available. Note that the terminological resource needs to have at least several thousand entries: the Eurovoc version used by Repar et al. (2019) contained 7,083 English-Slovene term pairs. We finally settled on the environmental thesaurus Gemet[4], which at the time had 3,721 Estonian-Russian term pairs. For the transliteration rules, we used the Python pip package transliterate [5] to generate the reverse dictionary-based features.

5 Results

Repar et al. (2019) ran a total of 10 parameter configurations. We selected three of those to test on the Estonian-Russian dataset. The first one is the configuration with a positive/negative ratio of 1:200 in the training set, which significantly improved recall compared to the reproduction of Aker et al. (2013b), the second one is the same configuration with additional term filtering, which was overall the best performing configuration in Repar et al. (2019), and the third one is the Cognates approach which should give greater weight to cognate words. As shown in Table 2, the overall results are considerably lower than the results in Repar et al. (2019), in particularly in terms of recall. One reason for this could be that the term filtering heuristics developed in Repar et al. (2019) may not work well for Estonian and Russian as they do for other languages. For example, 1.3 million candidate term pairs were constructed for the English-Slovene lan-

[2] https://doi.org/10.15155/9-00-0000-0000-0000-0002AL

[3] opus.nlpl.eu

[4] https://www.eionet.europa.eu/gemet/en/themes/

[5] https://pypi.org/project/transliterate/

No.	Config ET-RU	Training set size	Pos/Neg ratio	Precision	Recall	F-score
1	Training set 1:200	627,120	1:200	0.3237	0.2050	0.2510
2	Training set filtering 3	30,954	1:200	0.9000	0.0900	0.1636
3	Cognates approach	33,768	1:200	0.7313	0.0817	0.1469

Table 2: Results on the Estonian-Russian language pair. No. 1 presents the results of the configuration with a positive/negative ratio of 1:200 in the training set, no. 2 presents the results of the same configuration with additional term filtering, which was overall the best performing configuration in Repar et al. (2019), and No. 3 presents the results of the Cognates approach which should give greater weight to cognate words.

ET	RU	Evaluation
kontsert	концерт	exact match
kosmos	космос	exact match
majandus	экономика	exact match
juhiluba	водительские права	exact match
lõbustuspark	парк развлечений	exact match
unelmate pulm	свадьба	partial match
eesti mees	мужчина	partial match
indiaani horoskoop	гороскоп	partial match
hiina kapsas	капуста	partial match
hulkuvad koerad	собаки	partial match
eesti autospordi liit	эстонский футбольный союз	no match
Kalevi Kull	орел	no match
honda jazz	джаз	no match
tõnis mägi	гора	no match
linkin park	парк	no match

Table 3: Examples of exact, partial and no match tag pairs produced by the system.

guage pair and around one half of those were filtered out during the term filtering phase. On the other hand, only around 33,000 Estonian-Russian candidate pairs out of the total 627,000 survived the term fitering phase in these experiments. Another reason for the lower performance is likely the content of the language resources used to construct the features. Whereas Repar et al. (2019) use resources with similar content (EU legislation), here we have dictionary-based features constructed from a subtitle corpus and term pairs from an environmental thesaurus.

We then used the best performing configuration to try to align the Estonian and Russian tags from the dataset provided by Ekspress Meedia. The size of the dataset (59,632 Estonian tags and 6,198 Russian tags) and the fact that the system must test each possible pairing of source and target tags meant that the system generated around 370 million tag pair candidates which it then tried to classify as positive or negative. This task took more than two weeks to complete, but at the end it resulted in 4,989 positively classified Estonian-Russian tag pairs. A subset of these (500) were manually evaluated by a person with knowledge of both languages provided by Ekspress Meedia according to the following methodology:

- C: if the tag pair is a complete match
- P: if the tag pair is a partial match, i.e. when a multiword tag in one langauge is paired with a single word tag in the other language (e.g. eesti kontsert — концерт, or *Estonian concert — concert*)
- N: if the tag pair is a no match

Of the 500 positively classified tag pairs that were manually evaluated, 49% percent were deemed to be complete matches, a further 25% were evaluated as partial matches, and 26% were considered to be wrongly classified as positive tag pairs. The evaluator observed that "the most difficult thing was to separate people's names from toponyms, such as a famous local singer called "Tõnis Mägi", a district in Talinn called "Tõnismägi"

and a mountain named "Muna Mägi". More examples of exact, partial and no match alignments can be found in Table 3.

6 Conclusions and future work

In this paper, we reused an existing approach to terminology alignment by Repar et al. (2019) to align a set of Estonian and Russian tags provided by the media company Ekspress Meedia. The approach requires several bilingual resources to work and it was difficult to obtain relevant resources for the Estonian-Russian language pair. Given the domain of the tagset, i.e. news and media, the selected resources (subtitle translations and an environmental thesaurus) were less than ideal. Nevertheless, the approach provided respectable results with 74% of the positive tag pairs evaluated to be at least a partial match.

When asssessing the performance of the approach, one has to take into account the fact that the tagset is heavily unbalanced with almost 60,000 Estonian tags compared to a little over 6,000 Russian tags. This means that for many Estonian tags, a true equivalent was simply not available in the tagset.

For future work, we plan to integrate additional features into the algorithm, such as those based on novel neural network embeddings which may uncover additional hidden correlations between expressions in two different languages and may provide an alternative to large parallel corpora which are currently needed for the system for work. In terms of the Estonian and Russian language pair, additional improvements could be provided by taking into account the compound-like structure of many Estonian words. Finally, we will look into techniques that would allow us to pre-filter the initial list of tag pairs to reduce the total processing time.

7 Acknowledgements

The work was supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media).

References

Ahmet Aker, Monica Paramita, and Rob Gaizauskas. 2013a. Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 402–411, Sofia, Bulgaria. Association for Computational Linguistics.

Ahmet Aker, Monica Paramita, and Rob Gaizauskas. 2013b. Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 402–411.

Robert Gaizauskas, Ahmet Aker, and Robert Yang Feng. 2012. Automatic bilingual phrase extraction from comparable corpora. In *24th International Conference on Computational Linguistics*, pages 23–32.

Thorsten Joachims. 2002. *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.

Andraž Repar, Matej Martinc, and Senja Pollak. 2019. Reproduction, replication, analysis and adaptation of a term alignment approach. *Language Resources and Evaluation*, pages 1–34.

Ralf Steinberger, Andreas Eisele, Szymon Klocek, Spyridon Pilos, and Patrick Schlüter. 2013. DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC'2012)*.

Ralf Steinberger, Bruno Pouliquen, and Johan Hagman. 2002. Cross-lingual document similarity calculation using the multilingual thesaurus eurovoc. *Computational Linguistics and Intelligent Text Processing*, pages 101–121.

I Word-embedding based bilingual terminology alignment

Word-embedding based bilingual terminology alignment

Andraž Repar[1], Matej Martinc[2], Matej Ulčar[3], Senja Pollak[4]

[1]International Postgraduate School, Institut Jozef Stefan, Jamova 39, Ljubljana, Slovenia
[2] Institut Jozef Stefan, Jamova 39, Ljubljana, Slovenia
[3] Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, Ljubljana, Slovenia
[4] Institut Jozef Stefan, Jamova 39, Ljubljana, Slovenia
E-mail: andraz.repar@ijs.si, matej.martinc@ijs.si, matej.ulcar@fri.uni-lj.si, senja.pollak@ijs.si

The article will be published in the Proceedings of Elex 2021, the seventh biennial conference on electronic lexicography, held online, 5–7 July 2021.

Abstract

The ability to accurately align concepts between languages can provide significant benefits in many practical applications. In this paper, we extend a machine learning approach using dictionary and cognate-based features with novel cross-lingual embedding features using pretrained fastText embeddings. We use the tool VecMap to align the embeddings between Slovenian and English and then for every word calculate the top 3 closest word embeddings in the opposite language based on cosine distance. These alignments are then used as features for the machine learning algorithm. With one configuration of the input parameters, we managed to improve the overall F-score compared to previous work, while another configuration yielded improved precision (96%) at a cost of lower recall. Using embedding-based features as a replacement for dictionary-based features provides a significant benefit: while a large bilingual parallel corpus is required to generate the Giza++ word alignment lists, no such data is required for embedding-based features where the only required inputs are two unrelated monolingual corpora and a small bilingual dictionary from which the embedding alignments are calculated.

Keywords: terminology alignment; word embeddings; embeddings alignment; machine learning

1. Introduction

The ability to accurately align concepts between languages can provide significant benefits in many practical applications. For example, in terminology, terms can be aligned between languages to provide bilingual terminological resources, while in the news industry, keywords can be aligned to provide better news clustering or search in another language. Accurate bilingual resources can also serve as seed data for various other NLP tasks, such as multilingual vector space alignment.

Bilingual terminology alignment[1] is the process of aligning terms between two candidate term lists in two languages. The primary purpose of bilingual terminology extraction is to build a term bank - i.e. a list of terms in one language along with their equivalents in the other language. With regard to the input text, we can distinguish between alignment on the basis of a parallel corpus and alignment on the basis of a comparable corpus. For the translation industry, bilingual terminology extraction from parallel corpora is extremely relevant due to the large amounts of sentence-aligned parallel corpora available in the form of translation memories. Consequently, initial attempts at bilingual terminology extraction involved parallel input data (Kupiec, 1993; Daille et al., 1994; Gaussier, 1998), and the interest of the community continued until today. However, most parallel corpora are owned by private companies[2], such as language service providers, who consider them

[1] Note that bilingual terminology alignment has a narrower focus than *bilingual terminology extraction*, but the two terms are often used interchangeably in various papers. The latter covers extraction and alignment of terms between languages.

[2] However, some publicly available parallel corpora do exist. A good overview can be found at the OPUS web portal (Tiedemann, 2012).

to be their intellectual property and are reluctant to share them publicly. For this reason (and in particular for language pairs not involving English) considerable efforts have also been invested into researching bilingual terminology extraction from comparable corpora (Fung & Yee, 1998; Rapp, 1999; Chiao & Zweigenbaum, 2002; Cao & Li, 2002; Daille & Morin, 2005; Morin et al., 2008; Vintar, 2010; Bouamor et al., 2013; Hazem & Morin, 2016, 2017).

The approach designed by Aker et al. (2013) and replicated and adapted in Repar et al. (2019) served as the basis of our work. It was developed to align terminology between languages with the help of parallel corpora using machine-learning techniques. They use terms from the Eurovoc (Steinberger et al., 2002) thesaurus and train an SVM binary classifier (Joachims, 2002) (with a linear kernel and the trade-off between training error and margin parameter c = 10). The task of bilingual alignment is treated as a binary classification - each term from the source language S is paired with each term from the target language T and the classifier then decides whether the aligned pair is correct or incorrect. Aker et al. (2013) run their experiments on the 21 official EU languages covered by Eurovoc with English always being the source language (20 language pairs altogether). They evaluate the performance on a held-out term pair list from Eurovoc using recall, precision and F-measure for all 21 languages. Next, they propose an experimental setting for a simulation of a real-world scenario where they collect English-German comparable corpora of two domains (IT, automotive) from Wikipedia, perform monolingual term extraction using the system by Pinnis et al. (2012) followed by the bilingual alignment procedure described above and manually evaluate the results (using two evaluators). They report excellent performance on the held-out term list with many language pairs reaching 100% precision and the lowest recall being 65%. For Slovenian, which is of our main interest, the reported results were excellent with perfect or nearly perfect precision and good recall. The reported results of the manual evaluation phase were also good, with two evaluators agreeing that at least 81% of the extracted term pairs in the IT domain and at least 60% of the extracted term pairs in the automotive domain can be considered exact translations. Repar et al. (2019) tried to reproduce their approach and after initially having little success they were at the end able to achieve comparable results with precision exceeding 90% and recall over 50%.

Despite the problem of bilingual term alignment lending itself well to the binary classification task, there have been relatively few approaches utilizing machine learning. Similar to Aker et al. (2013), Baldwin & Tanaka (2004) generate corpus-based, dictionary-based and translation-based features and train an SVM classifier to rank the translation candidates. Note that they only focus on multi-word noun phrases (noun + noun). A similar approach, again focusing on noun phrases, is also described by Cao & Li (2002). Finally, Nassirudin & Purwarianti (2015) also reimplement Aker et al. (2013) for the Indonesian-Japanese language pair and further expand it with additional statistical features.

This paper is organized as follows: the present section introduces the problem and related work, Section 2 describes the datasets used for the experiments, Section 3 lists the features used in the machine learning process, Section 4 contains a description of the experiments and lists their results and section 5 provides the conclusion.

2. Resources

The approach described in this paper requires four types of resources. The first two are the same as in Aker et al. (2013) and Repar et al. (2019), whereas the third and fourth resources are required for the additional experiments conducted for this paper:

- aligned term pairs in two languages that serve as training data
- a parallel corpus to generate a Giza++ word alignment list
- pretrained embeddings in two languages
- a (small) bilingual dictionary

We create term pairs from the Eurovoc (Steinberger et al., 2002) thesaurus, which at the time of Repar et al. (2019) consisted of 7,083[3] terms, by pairing Slovenian terms with English ones. The test set consisted of 600 positive (correct) term pairs—taken randomly out of the total 7,083 Eurovoc term pairs—and around 1.3 million negative pairs which were created by pairing each source term with 200 distinct incorrect random target terms. Aker et al. (2013) argue that this was done to simulate real-world conditions where the classifier would be faced with a larger number of negative pairs and a comparably small number of positive ones. The 600 positive term pairs were further divided into 200 pairs where both (i.e. source and target) terms were single words, 200 pairs with a single word only on one side and 200 pairs with multiple-word terms on both sides. The remaining positive term pairs (approximately 6,200) were used as training data along with additional 6,200 negative pairs. These were constructed by taking the source side terms and pairing each source term with one target term (other than the correct one). Using Giza++, we created source-to-target and target-to-source word alignment dictionaries based on the DGT translation memory (Steinberger et al., 2013). The resulting dictionary entries consist of the source word s, its translation t and the number indicating the probability that t is an actual translation of s. To improve the performance of the dictionary-based features, the following entries were removed from the dictionaries:

- entries where probability is lower then 0.05
- entries where the source word was less than 4 characters and the target word more than 5 characters long and vice versa in order to avoid translations of stop word to content words)

In addition to the resources described above, we used fastText (Bojanowski et al., 2016) pre-trained word embedding vectors to calculate distances (or similarities) between terms. We aligned monolingual fastText embeddings using the VecMap (Artetxe et al., 2018) tool which can align embeddings with the help of a small bilingual dictionary. We used a bilingual dictionary compiled from two sources: single word terms from Eurovoc and Wiktionary entries extracted using wikt2dict tool (Acs, 2014). Using the aligned embedding vectors, we then calculated cosine distances between all words present in Eurovoc terms in one language and all words present in Eurovoc terms in the other language.

Using the fastText-based lists of aligned words, we created 3-tuples[4] of most similar - based on cosine similarity - source-to-target and target-to-source words, such as:

[3] While new terms are constantly added to Eurovoc, we decided not to use them to allow for better comparison between the approaches

[4] This number was determined experimentally.

- ksenofobija ['xenophobia', '0.744'], ['racism', '0.6797'], ['anti-semitism', '0.654']
- ženska ['woman', '0.7896'], ['women', '0.73'], ['female', '0.722']

where the tuple contains the source language word along with their three most likely corresponding words in the target language and their cosine similarities. The 3-tuples of most similar words were used to construct additional features for the machine learning algorithm.

3. Feature construction

The updated approach in this paper uses three types of features that express correspondences between the words (composing a term) in the target and source language. The dictionary and cognate-based features are same as in Repar et al. (2019), while embedding-based features are newly developed. The three feature types are (for a detailed description see Table 1):

- 7 dictionary-based (using Giza++) features which take advantage of dictionaries created from large parallel corpora of which 6 are direction-dependent (source-to-target or target-to-source) and 1 direction-independent - resulting in altogether 13 features
- 7 cognate-based features (on the basis of Gaizauskas et al. (2012)) which utilize string-based word similarity between languages
- 5 cognate-based features using specific transliteration rules which take into account the differences in writing systems between two languages: e.g. Slovenian and English. Transliteration rules were created for both directions (source-to-target and target-to-source) separately and cognate-based features were constructed for both directions - resulting in additional 10 cognate-based features with transliteration rules. The following transliteration rules were used: *x:ks, y:j, w:v, q:k* for English to Slovenian and *č:ch, š:sh, ž:zh* for Slovenian to English
- 5 direction-dependent combined[5] features where the term pair alignment is correct if either the dictionary or the cognate-based method returns a positive result - resulting in a total of 10 combined features
- 12 novel direction-dependent embedding-based features utilizing fastText embeddings - resulting in a total of 24 features
- 5 novel combined features constructed in the same manner as the exisiting combined features but replacing Giza++ word lists with fastText-based lists of top 3 aligned words - resulting in a total of 10 novel combined features
- 3 term length features: sourceTargetLengthMatch, sourceTermLength, targetTermLength

To match words with morphological differences, we do not perform direct string matching but utilize Levenshtein Distance. Two words were considered equal if the Levenshtein Distance Levenshtein (1966) was equal or higher than 0.95.

[5] For combined features, a word is considered as covered if it can be found in the corresponding set of Giza++ translations or if one of the cognate-based measures (Longest Common Subsequence, Longest Common Substring, Levensthein Distance, Needleman-Wunsch Distance, Dice) is 0.70 or higher (set experimentally by Aker et al. (2013))

Feature	Cat	Description
isFirstWordTranslated	Dict	Checks whether the first word of the source term is a translation of the first word in the target term (based on the Giza++ dictionary)
isLastWordTranslated	Dict	Checks whether the last word of the source term is a translation of the last word in the target term
percentageOfTranslatedWords	Dict	Ratio of source words that have a translation in the target term
percentageOfNotTranslatedWords	Dict	Ratio of source words that do not have a translation in the target term
longestTranslatedUnitInPercentage	Dict	Ratio of the longest contiguous sequence of source words which has a translation in the target term (compared to the source term length)
longestNotTranslatedUnitInPercentage	Dict	Ratio of the longest contiguous sequence of source words which do not have a translation in the target term (compared to the source term length)
Longest Common Subsequence Ratio	Cogn	Measures the longest common non-consecutive sequence of characters between two strings
Longest Common Substring Ratio	Cogn	Measures the longest common consecutive string (LCST) of characters that two strings have in common
Dice similarity	Cogn	2*LCST / (len(source) + len(target))
Needlemann-Wunsch distance	Cogn	LCST / min(len(source), len(target))
isFirstWordCognate	Cogn	A binary feature which returns True if the longest common consecutive string (LCST) of the first words in the source and target terms divided by the length of the longest of the two words is greater than or equal to a threshold value of 0.7 and both words are longer than 3 characters
isLastWordCognate	Cogn	A binary feature which returns True if the longest common consecutive string (LCST) of the last words in the source and target terms divided by the length of longest of the two words is greater than or equal to a threshold value of 0.7 and both words are longer than 3 characters
Normalized Levensthein distance (LD)	Cogn	1 - LD / max(len(source), len(target))
isFirstWordCovered	Comb	A binary feature indicating whether the first word in the source term has a translation or transliteration in the target term
isLastWordCovered	Comb	A binary feature indicating whether the last word in the source term has a translation or transliteration in the target term
percentageOfCoverage	Comb	Returns the percentage of source term words which have a translation or transliteration in the target term
percentageOfNonCoverage	Comb	Returns the percentage of source term words which have neither a translation nor transliteration in the target term
difBetweenCoverageAndNonCoverage	Comb	Returns the difference between the last two features
isFirstWordMatch	Emd	Checks whether the first word of the source term is the most likely translation of the first word in the target term (based on the aligned embeddings)
isLastWordMatch	Emd	Checks whether the last word of the source term is the most likely translation of the last word in the target term (based on the aligned embeddings)
percentageOfFirstMatchWords	Emb	Ratio of source words that have a first match (i.e. first position in the 3-tuple) in the target term
percentageOfNotFirstMatchWords	Emb	Ratio of source words that do not have a first match (i.e. first position in the 3-tuple) in the target term
longestFirstMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which has a first match (first position in the 3-tuple) in the target term (compared to the source term length)
longestNotFirstMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which do not have a first match (first position in the 3-tuple) in the target term (compared to the source term length)
isFirstWordTopnMatch	Emd	Checks whether the first word of the source term is in the 3-tuple of most likely translations of the first word in the target term (based on the aligned embeddings)

isLastWordTopnMatch	Emd	Checks whether the first word of the source term is not in the 3-tuple of most likely translations of the first word in the target term (based on the aligned embeddings)
percentageOfTopnMatchWords	Emb	Ratio of source words that have a match (i.e. any position in the 3-tuple) in the target term
percentageOfNotTopnMatchWords	Emb	Ratio of source words that do not have a match (i.e. any position in the 3-tuple) in the target term
longestTopnMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which has a match (any position in the 3-tuple) in the target term (compared to the source term length)
longestNotTopnMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which do not have a match (any position in the 3-tuple) in the target term (compared to the source term length)
isFirstWordCoveredEmbeddings	Comb	A binary feature indicating whether the first word in the source term has a match (any position in the 3-tuple) or transliteration in the target term
isLastWordCoveredEmbeddings	Comb	A binary feature indicating whether the last word in the source term has a match (any position in the 3-tuple) or transliteration in the target term
percentageOfCoverageEmbeddings	Comb	Returns the percentage of source term words which have a match (any position in the 3-tuple) or transliteration in the target term
percentageOfNonCoverageEmbeddings	Comb	Returns the percentage of source term words which do not have a match (any position in the 3-tuple) or transliteration in the target term
diffBetweenCoverageAnd-NonCoverageEmbeddings	Comb	Returns the difference between the last two features

Figure 1: Features used in the experiments. Note that some features are used more than once because they are direction-dependent.

4. Experimental setup and results

The constructed features were then used to train an SVM binary classifier (Joachims, 2002) (with a linear kernel and the trade-off between training error and margin parameter c = 10). We selected three configurations from Repar et al. (2019) for comparison:

- Training set 1:200: a very unbalanced train set (ratio of 1:200 between positive and negative examples [6]) greatly improves the precision of the classifier at a cost of somewhat lower recall, when compared to a balanced train set or a less unbalanced train set (e.g., ratio of 1:10 between positive and negative examples).
- Training set filtering 3: In Repar et al. (2019), we have performed an error analysis and found that many incorrectly classified term pairs are cases of partial translation where one unit in a multi-word term has a correct Giza++ dictionary translation in the corresponding term in the other language. Based on the problem of partial translations, leading to false positive examples, we focused on the features that would eliminate such partial translations from the training set. After a systematic experimentation, we noticed that we can drastically improve precision if we only keep positive term pairs with the following feature values: isFirstWordTranslated = True, isLastWordTranslated = True, percentageOfCoverage > 0.66, isFirstWordTranslated-reversed = True, isLastWordTranslated-reversed = True, percentageOfCoverage-reversed > 0.66.

[6] 1:200 imbalance ratio was the largest imbalance we tried, since the testing results indicated that no further gains could be achieved by further increasing the imbalance.

- Cognates: the dataset is additionally filtered according to the following criteria: isFirstWordCognate = True and isLastWordCognate = True, isFirstWordTranslated = True and isLastWordCognate = True, isFirstWordCognate = True and isLastWordTranslated = True and we also use a Gaussian kernel instead of the linear one, since this new dataset structure represents a classic "exclusive or" (XOR) problem which a linear classifier is unable to solve.

The selection was made based on our experience and previous work with this approach. The three selected configurations were among the best performing in previous experiments and we believed they had the highest potential for improvement. For a complete description of the decisions that led to these configurations, please refer to Repar et al. (2019).

No.	Config EN-SL	Training set size	Pos/Neg ratio	Precision	Recall	F-score
	Dictionary-based and cognate-based features					
1	Training set 1:200	1,303,083	1:200	0.4299	**0.7617**	0.5496
2	Training set filtering 3	645,813	1:200	0.9342	0.4966	0.6485
3	Cognates approach	672,345	1:200	0.8732	0.5167	0.6492
	Dictionary-based, embedding-based and cognate-based features					
1	Training set 1:200	1,303,083	1:200	0.5375	0.680	0.6004
2	Training set filtering 3	695,058	1:200	0.8170	0.5133	0.6305
3	Cognates approach	706,113	1:200	0.8991	0.5200	**0.6589**
	Embedding-based and cognate-based features only					
1	Training set 1:200	1,303,083	1:200	0.3232	0.4967	0.3916
2	Training set filtering 3	322,605	1:200	0.9545	0.2450	0.3899
3	Cognates approach	394,362	1:200	**0.9618**	0.3617	0.5242

Table 2: Results on the English-Slovenian term pair.

First, we simply added the new embedding-based features to the dataset to see if they improved the overall performance. Later, we removed the dictionary-based features from the dataset to see whether the novel embedding-based features could replace them without a major impact to the performance. As can be observed from Table 2, the results are a mixed bag when using all available features. Without any training set filtering, the new features improve precision at the expense of recall, but are less effective when filtering is applied. Nevertheless, when we use additional trainset filters for the Cognates approach, we can observe a slight increase in both precision and recall resulting in the overall highest F-score. When we use only embedding-based and cognate-based features, which would be beneficial for language pairs without access to large parallel corpora needed to create Giza++ word alignments, there is a significant drop in recall in all cases, but precision actually increases when trainset filtering is applied and the Cognates approach achieves the overall best precision.

5. Conclusion

In this paper, we continued our experiments on bilingual terminology alignment using a machine learning approach by adding new features based on fastText word embedding

vectors. We took advantage of the availability of large pre-trained datasets by Bojanowski et al. (2016), and a cross-lingual word embedding mapping tool Vecmap by Artetxe et al. (2018) to create word alignment dictionaries similar to the output of traditional word alignment tools, such as Giza++ (Och & Ney, 2003). The single most important advantage of this approach is that while Giza++ requires a large parallel corpus, fastText vectors are trained on monolingual data and Vecmap needs only a (much smaller) bilingual dictionary. Bilingual dictionaries are readily available for many language pairs via Wiktionary (Acs, 2014).

The experiments showed that the new features can have a positive impact on F-score (depending on the configuration), but precision was somewhat lower compared to when we were using only Giza++ features. When we removed Giza++ features and using only the new embedding-based features (alongside cognate features which are based on word similarity and require no pre-existing bilingual data), we observed somewhat lower recall and a bit higher precision. This means that the embedding-based features can be used instead of Giza++ features for language pairs where no large parallel bilingual corpora is available.

In terms of future work, we plan on creating additional features using contextual embeddings, such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018), which could potential help us improve recall, and explore more granular and detailed training set filtering techniques. We also plan to expand the experiments and test other configurations in a more systematic way.

6. Acknowledgements

The work was supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). We also acknowledge the project the financial support from the Slovenian Research Agency for research core funding for the programme Knowledge Technologies (No. P2-0103) and the project TermFrame - Terminology and Knowledge Frames across Languages (No. J6-9372).

7. References

Acs, J. (2014). Pivot-based multilingual dictionary building using Wiktionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation LREC*.

Aker, A., Paramita, M. & Gaizauskas, R. (2013). Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1. pp. 402–411.

Artetxe, M., Labaka, G. & Agirre, E. (2018). Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Baldwin, T. & Tanaka, T. (2004). Translation by machine of complex nominals: Getting it right. In *Proceedings of the Workshop on Multiword Expressions: Integrating Processing*. pp. 24–31.

Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T. (2016). Enriching Word Vectors with Subword Information. *arXiv preprint arXiv:1607.04606*.

Bouamor, D., Semmar, N. & Zweigenbaum, P. (2013). Context Vector Disambiguation for Bilingual Lexicon Extraction from Comparable Corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. pp. 759–764.

Cao, Y. & Li, H. (2002). Base Noun Phrase Translation Using Web Data and the EM Algorithm. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*. pp. 1–7.

Chiao, Y.C. & Zweigenbaum, P. (2002). Looking for Candidate Translational Equivalents in Specialized, Comparable Corpora. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 2*. pp. 1–5.

Daille, B., Gaussier, E. & Langé, J.M. (1994). Towards Automatic Extraction of Monolingual and Bilingual Terminology. In *Proceedings of the 15th Conference on Computational Linguistics - Volume 1*. pp. 515–521.

Daille, B. & Morin, E. (2005). French-English Terminology Extraction from Comparable Corpora. In *Natural Language Processing – IJCNLP 2005*. pp. 707–718.

Devlin, J., Chang, M.W., Lee, K. & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.

Fung, P. & Yee, L.Y. (1998). An IR Approach for Translating New Words from Nonparallel, Comparable Texts. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*. pp. 414–420.

Gaizauskas, R., Aker, A. & Yang Feng, R. (2012). Automatic bilingual phrase extraction from comparable corpora. In *24th International Conference on Computational Linguistics*. pp. 23–32.

Gaussier, E. (1998). Flow Network Models for Word Alignment and Terminology Extraction from Bilingual Corpora. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*. pp. 444–450.

Hazem, A. & Morin, E. (2016). Efficient Data Selection for Bilingual Terminology Extraction from Comparable Corpora. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. pp. 3401–3411.

Hazem, A. & Morin, E. (2017). Bilingual Word Embeddings for Bilingual Terminology Extraction from Specialized Comparable Corpora. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. pp. 685–693.

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

Kupiec, J. (1993). An algorithm for finding noun phrase correspondences in bilingual corpora. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*. pp. 17–22.

Levenshtein, V.I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10, p. 707.

Morin, E., Daille, B., Takeuchi, K. & Kageura, K. (2008). Brains, Not Brawn: The Use of Smart Comparable Corpora in Bilingual Terminology Mining. *ACM Trans. Speech Lang. Process.*, 7(1), pp. 1:1–1:23.

Nassirudin, M. & Purwarianti, A. (2015). Indonesian-Japanese term extraction from bilingual corpora using machine learning. In *Advanced Computer Science and Information Systems (ICACSIS), 2015 International Conference on*. pp. 111–116.

Och, F.J. & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1), pp. 19–51.

Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. & Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. of NAACL*.

Pinnis, M., Ljubešić, N., Stefanescu, D., Skadina, I., Tadić, M. & Gornostaya, T. (2012). Term extraction, tagging, and mapping tools for under-resourced languages. In *Proceedings of the 10th Conference on Terminology and Knowledge Engineering (TKE 2012), June*. pp. 20–21.

Rapp, R. (1999). Automatic Identification of Word Translations from Unrelated English and German Corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*. pp. 519–526.

Repar, A., Martinc, M. & Pollak, S. (2019). Reproduction, replication, analysis and adaptation of a term alignment approach. *Language Resources and Evaluation*, pp. 1–34.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S. & Schlüter, P. (2013). DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC'2012)*.

Steinberger, R., Pouliquen, B. & Hagman, J. (2002). Cross-lingual document similarity calculation using the multilingual thesaurus eurovoc. *Computational Linguistics and Intelligent Text Processing*, pp. 101–121.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In N.C.C. Chair), K. Choukri, T. Declerck, M.U. Dogan, B. Maegaard, J. Mariani, J. Odijk & S. Piperidis (eds.) *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey: European Language Resources Association (ELRA).

Vintar, Š. (2010). Bilingual term recognition revisited: The bag-of-equivalents term alignment approach and its evaluation. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 16(2), pp. 141–158.