DeLFT and *entity-fishing*: Tools for CLEF HIPE 2020 Shared Task

Tanti Kristanti[0000−0003−3524−020X] and Laurent Romary[0000−0002−0756−0508]

Inria, Paris
{tanti.kristanti,laurent.romary}@inria.fr

Abstract. This article presents an overview of approaches and results during our participation in the CLEF HIPE 2020 NERC-COARSE-LIT and EL-ONLY tasks for English and French. For these two tasks, we use two systems: 1) DeLFT, a Deep Learning framework for text processing; 2) *entity-fishing*, generic named entity recognition and disambiguation service deployed in the technical framework of INRIA.

Keywords: entity recognition · entity linking · machine learning · deep learning

1 Introduction

Named Entity Recognition (NER) refers to the task of identifying text spans that mention named entities and classifying them into predefined classes (e.g., person, location, organization). Whereas, Entity Linking (EL) refers to the task of detecting textual entity mentions and matching them to corresponding entries within knowledge bases (e.g., Wikipedia, Wikidata). Over the past few years, traditional machine learning-based (i.e., rule-based, unsupervised, and feature-based supervised learning) approaches have evolved into deep learning (DL) approaches, yielding state-of-the-art performances [7, 8, 21]. For our participation in the CLEF HIPE 2020 shared task, we use two different systems that implement non-neural machine learning (ML) and DL approaches.

In HIPE, a shared task dedicated to the evaluation of named entity (NE) processing on historical newspapers in French, German, and English [5], we participated in two different NERC-COARSE-LIT and EL-ONLY tasks by using two systems: DeLFT and *entity-fishing*. Deep Learning Framework for Text (DeLFT) is a framework for text processing, which re-implements standard state-of-the-art DL architectures. Meanwhile, *entity-fishing* is a generic named entity recognition and disambiguation (NERD) system, which applies supervised machine learning based on Random Forest and Gradient Tree Boosting exploiting various features.

Both DeLFT[1] and *entity-fishing*[2] are open-source systems under an Apache 2 license. Codes, models, and resources that are publicly available through the Github repository allow users and contributors, including us, to use existing services and models and to contribute to system and model improvements and tests. With the use of DeLFT, our goal is to re-build DL-based models for recognizing mentions within English and French HIPE historical corpus belonging to Person (pers), Location (LOC), Organization (ORG), Product (PROD), and Time (TIME) classes. Meanwhile, with *entity-fishing* service, we use the service to disambiguate provided mentions within French and English HIPE data against Wikidata entries.

2 Tools

This section provides a general description of the two systems we use, DeLFT and *entity-fishing*. For a better understanding of technical discussions, it is advisable to refer directly to their repositories and documentation.

2.1 DeLFT

Deep Learning Framework for Text (DeLFT) is an open-source framework for text processing, including sequence labeling (e.g., NE tagging) and text classification problems. This Keras and TensorFlow framework re-implements standard state-of-the-art DL architectures for text processing.

DeLFT supports many DL architectures (e.g., Bidirectional LSTMs and Conditional Random Fields [8], Bidirectional LSTM and Convolutional Neural Networks [2], Bidirectional Gated Recurrent Unit [15]) and contextualized embeddings (e.g., ELMo, BERT). For using the desired pre-trained word embeddings, we need to provide them from the source separately. DeLFT then loads and manages these embeddings by compiling at the very first access to be stored in a database.

2.2 *entity-fishing*

entity-fishing is a generic NERD system against Wikidata. Deployed as part of the French national infrastructure Huma-Num[3], the service provides a standardized interface, open and flexible architecture, allowing easy deployment, including in digital humanities contexts. Initiated in the context of the EU FP7 Cendari project from 2013 to 2016, *entity-fishing* aimed at setting up a digital research environment for historians of the medieval and WWI periods to access archival contents and acquire numerous assets or entities information [6].

In general, *entity-fishing* has three phases: language identification, mention recognition, and entity resolution. First, language identification is necessary for

[1] https://github.com/kermitt2/delft
[2] https://github.com/kermitt2/entity-fishing
[3] https://www.huma-num.fr/

selecting appropriate utilities for text processing (e.g., tokenizer, sentence segmentation) and a specific Wikipedia from the knowledge base. Second, mention recognition has the responsibility to extract entity mentions from the input. To support the generic nature, even though prepared with a set of recognizers, *entity-fishing* provides the possibility for users to extend with additional ones.

entity-fishing supports traditional mention extractors: named entity recognition, Wikipedia lookup, acronym extraction. For NER, *entity-fishing* uses grobid-ner. grobid-ner[4] is a library for processing texts, extracting named entities, and classifying these entities into 27 classes (e.g., person, location, media, organization, period) using a Conditional Random Field (CRF) statistical model. Meanwhile, Wikipedia lookup is complementary to the machine learning NER approach. The lookup attempts to find all mentions that correspond to either a title or an anchor in Wikipedia using an N-gram-based matching approach. For the acronyms, *entity-fishing* treats them as mentions and uses the base for disambiguation. The resolved entity is then further propagated in the text for each occurrence of the acronym. The result of the mention recognition step is an aggregated list of objects containing raw values from the original text, their actual positions, and NER classes (within the 27 classes).

Lastly, entity resolution is the process of matching entity mentions to their corresponding Wikidata entries. The entity resolution has three stages: candidate generation, candidate ranking, candidate selector. In the candidate generation phase, each mention has a list of possible candidates for the disambiguation. Then, in the candidate ranking, each candidate is assigned a confidence score calculated as regression probability using various features.

2.3 Auxiliary Resources

We use external datasets and embeddings in addition to those provided by the HIPE organizers.

Datasets The HIPE corpus consists of training, dev, and test datasets for each task and language. However, since for English, HIPE does not provide the training data, we use a pre-trained CoNLL-2012 (based on Ontonotes 5.0) [17] model and test the model with the HIPE test data.

Moreover, motivated by the promising French model results published by DeLFT, we use the annotated [19] French TreeBank (FTB) corpus and the HIPE data to re-build and benchmark the NER French model. This French journalistic genre corpus from the year 1990 is the annotated 2007 version of the FTB treebank containing the span, the literal type, sometimes completed with a subtype, and Aleda unique identifier of each mention. They use seven basic classes: Person, Location, Organization, Company, Product, POI (Point of Interest), and FictionChar (fictional character). FTB corpus contains 11,636 manually annotated mentions with the distribution of 3,761 location names, 3,357 company

[4] https://github.com/kermitt2/grobid-ner

names, 2,381 organization names, 2,025 person names, 67 product names, 29 fictional character names, and 15 POIs.

Word Embeddings We use various static word embeddings: Global Vectors for Word Representation (GloVe) [14], English fastText Common Crawl [1, 11], and French Wikipedia fastText.[5] We also use ELMo [16] contextualized word representations for English[6] and French[7].

Table 1. Comparison of DeLFT NER models with various feature sets and other published systems.

Model	**CoNLL-2003** [20]	**Ontonotes 5.0** [17]	**FTB** [19]
	F1-score		
DeLFT models [3]			
CoNLL-2003-BiLSTM-CRF + GloVe	91.35	-	-
CoNLL-2003-BiLSTM-CRF + GloVe + ELMo	92.71	-	-
CoNLL-2003-BiLSTM-CRF + GloVe + ELMo + valid set	**93.09**	-	-
CoNLL-2003-BiLSTM-CNN-CRF + GloVe	91.07	-	-
CoNLL-2003-BiLSTM-CNN-CRF + GloVe + ELMo	92.57	-	-
CoNLL-2003-BiLSTM-CNN-CRF + GloVe + ELMo + valid set	93.04	-	-
CoNLL-2003-BiLSTM-CNN + GloVe	89.47	-	-
CoNLL-2003-BiLSTM-CNN + GloVe + ELMo	92.00	-	-
CoNLL-2003-BiLSTM-CNN + GloVe + ELMo + valid set	92.16	-	-
CoNLL-2003-BiGRU-CRF + GloVe	90.72	-	-
CoNLL-2003-BiGRU-CRF + GloVe + ELMo	92.44	-	-
CoNLL-2003-BiGRU-CRF + GloVe + ELMo + valid set	92.71	-	-
CoNLL-2003-BERT-base	90.90	-	-
CoNLL-2003-BERT-base + CRF	91.20	-	-
CoNLL-2012-BiLSTM-CRF + fastText	-	87.01	-
CoNLL-2012-BiLSTM-CRF + fastText + ELMo	-	**89.01**	-
FTB-BiLSTM-CRF + fastText	-	-	87.45
FTB-BiLSTM-CRF + fastText + ELMo	-	-	**89.23**
neural achitectures			
Lample, et al. (2016) [8]	90.94	-	-
Ma and Hovy (2016) [10]	91.21	-	-
Chiu and Eric (2016) [2]	91.62	86.28	-
Peters, et al. (2018) [16]	92.22	-	-
Devlin, et al. (2018) [4]	92.80	-	-
non neural achitectures			
Ratinov and Roth (2009) [18]	90.80	-	-
Passos, et al. (2014) [13]	90.90	82.30	-
Luo, et al. (2015) [9]	89.90	-	-
Luo, et al. (2015) + linking [9]	91.20	-	-

3 Benchmarking NER Models

Machine learning approaches have dominated NER, but the trend is towards neural network architectures that achieve state-of-the-art performances. For this

[5] https://s3-us-west-1.amazonaws.com/fasttext-vectors/wiki.fr.vec
[6] https://s3-us-west-2.amazonaws.com/allennlp/models/ELMo/
[7] https://traces1.inria.fr/oscar/files/models/cc/fr.zip

reason, we compare several published NER systems as well as DeLFT pre-trained models against various corpora (i.e., CoNLL-2003, CoNLL-2012, FTB) and present them in Table 1.

[18] non-neural machine learning system achieves a 90.80 F1-score on CoNLL-2003. [13] improves with 90.90 on CoNLL-2003 and 82.30 on Ontonotes 5.0. Although [9] exceeds the previous achievements with their NERD system, which pushes the F1-score to 91.20. Nevertheless, their NER pure system reaches 89.90.

Meanwhile, for neural architectures, [8] reaches a 90.94 F1-score on CoNLL-2003, then [10] improves the results. [2] reports an F1-score of 91.62 on CoNLL-2003 and 86.28 on OntoNotes 5.0. [16] ELMo enhanced bidirectional LSTM with a CRF layer (BiLSTM-CRF) achieves an averaged F1-score of 92.22 over five runs. [4] has a competitive performance with state-of-the-art systems where its $\text{BERT}_{\text{LARGE}}$ fine-tuning approach tested on CoNLL-2003 reaches 92.80.

DeLFT has reimplemented neural architectures for NER [3]. Table 1 presents reported best F1-scores over ten runs for the English model using CoNLL-2003 and CoNLL-2012 and the French model using the FTB corpus.

The model trained with CoNLL-2003 within the BiLSTM-CRF architecture and GloVe word embedding, tested against the test set, achieves a 91.35 F1-score. The result is improving with GloVe combined with ELMo. Within BERT architecture and the CRF activation layer for fine-tuning, the model achieves an average of 91.20 F1-score. The best F1-score on CoNLL-2003 is 93.09 when using both train and validation datasets within a BiLSTM-CRF architecture, coupled with GloVe and ELMo embeddings. Meanwhile, the CoNLL-2012-based model within the BiLSTM-CRF architecture and the fastText embedding achieves an F1-score of 87.01. The involvement of ELMo increases the score by 2 points to 89.01.

The French model trained with the FTB corpus within the BiLSTM-CRF architecture and French Wikipedia fastText reaches an 87.45 F1-score. Meanwhile, with the use of French ELMo, the score is improving into 89.23.

From these results, we learn that DL-based systems have better performance than conventional machine learning systems. The use of contextualized word embeddings within the BiLSTM-CRF architecture improves the scores. The results in the CoNLL-2003 column also show that ELMo-based models give better F1-score than BERT-based models.

4 Work Phases

In general, the HIPE shared task contains two tasks:

1. Named Entity Recognition and Classification (NERC): the recognition and classification of entity mentions with predefined high-level (i.e., pers, org, prod, loc, time), finer-grained, or nested entity classes.
2. Named Entity Linking (NEL): the task of matching identified entity mentions to Wikidata entries, with or without prior knowledge of mention types and boundaries.

4.1 Named Entity Recognition and Classification (NERC)

Although the English model built with the CoNLL-2003 dataset is promising, this model does not support the Time (Date) entities. Moreover, since HIPE does not provide training data for English, we use a CoNLL-2012 pre-trained model within the BiLSTM-CRF architecture, and ELMo contextualized word embeddings. For the French model, we enrich the French HIPE (i.e., the version 1.2 train and dev) dataset with annotated FTB data.

For training the models, we follow the default hyper-parameters [8] as applied for other pre-trained sequence labeling models in DeLFT, except for the batch size and maximum epoch, we follow as indicates here.[9]

Challenges Combining data from different environments poses challenges, particularly with the reason of different NE class definitions as well as annotation guidelines. CoNLL-2012 define 18 classes. FTB corpus comes with seven NE classes. Meanwhile, HIPE uses five classes.

HIPE annotates absolute dates without months and hours, which confirms to the CoNLL-2012 DATE class. Furthermore, the HIPE Location (loc) entities suites with those belonging to CoNLL-2012 FAC (i.e., buildings, airports, highways, bridges), GPE (i.e., countries, cities, states), and LOCATION (i.e., non-GPE locations, mountain ranges, bodies of water) entities. It's also challenging to search the equivalence of the HIPE PROD entities, which we understand as media products since CoNLL-2012 classifies them in the ORG class.

Experiments We benchmarked the French NER trained with the HIPE data (i.e., train and dev v-1.2) only and the HIPE plus additional FTB data. The only HIPE model achieved an F1-score of 85.71 on the dev set. Meanwhile, the enriched model performs better with an increase of 3 scores into 88.46.

4.2 Named Entity Linking

For the NEL task, we use entity extraction and disambiguation services provided by *entity-fishing*. There are several ways to access these services; however, the most straightforward way is to use the service RESTful web services.[10]

First, we collect the text from the HIPE data. Then, we include this text as part of the JSON input query. The *entity-fishing* query processing service takes as input a JSON structured query and returns the JSON query enriched with a list of identified and, when possible, disambiguated entities. The JSON query format for the response file is identical to the input query. The client must respect the JSON query format, which is as follow:

[8] https://github.com/kermitt2/delft/blob/master/delft/sequenceLabelling/config.py
[9] https://github.com/kermitt2/delft/blob/master/nerTagger.py
[10] https://nerd.readthedocs.io/en/latest/restAPI.html

```
{
    "text": "The text to be processed.",
    "shortText": "term1 term2 ...",
    "termVector": [
        {
            "term": "term1",
            "score": 0.3
        },
        {
            "term": "term2",
            "score": 0.1
        }
    ],
    "language": {
        "lang": "en"
    },
    "entities": [],
    "mentions": ["ner","wikipedia"],
    "nbest": 0,
    "sentence": false,
    "customisation": "generic",
    "processSentence": [],
    "structure": "grobid"
}
```


Table 2. NERC-COARSE-LIT and EL-ONLY results compared to the best system and the baseline. Our results are highlighted.

Lang	Team	Evaluation	Precision	Recall	F1
EN	L3i	NE-COARSE-LIT-micro-strict	0.623	0.641	0.632
EN	**Inria-DeLFT**	**NE-COARSE-LIT-micro-strict**	**0.461**	**0.606**	**0.524**
EN	Baseline	NE-COARSE-LIT-micro-strict	0.531	0.327	0.405
EN	L3i	NE-COARSE-LIT-micro-fuzzy	0.794	0.817	0.806
EN	**Inria-DeLFT**	**NE-COARSE-LIT-micro-fuzzy**	**0.568**	**0.746**	**0.645**
EN	Baseline	NE-COARSE-LIT-micro-fuzzy	0.736	0.454	0.562
FR	L3i	NE-COARSE-LIT-micro-strict	0.831	0.849	0.84
FR	Baseline	NE-COARSE-LIT-micro-strict	0.693	0.606	0.646
FR	**Inria-DeLFT**	**NE-COARSE-LIT-micro-strict**	**0.605**	**0.675**	**0.638**
FR	L3i	NE-COARSE-LIT-micro-fuzzy	0.912	0.931	0.921
FR	**Inria-DeLFT**	**NE-COARSE-LIT-micro-fuzzy**	**0.755**	**0.842**	**0.796**
FR	Baseline	NE-COARSE-LIT-micro-fuzzy	0.825	0.721	0.769
EN	**Inria-DeLFT**	**NEL-LIT-micro-fuzzy-@1**	**0.633**	**0.685**	**0.658**
EN	L3i	NEL-LIT-micro-fuzzy-@1	0.593	0.593	0.593
EN	aidalight-baseline	NEL-LIT-micro-fuzzy-@1	0.506	0.506	0.506
FR	L3i	NEL-LIT-micro-fuzzy-@1	0.64	0.638	0.639
FR	**Inria-DeLFT**	**NEL-LIT-micro-fuzzy-@1**	**0.585**	**0.65**	**0.616**
FR	aidalight-baseline	NEL-LIT-micro-fuzzy-@1	0.502	0.495	0.498

5 Results

Table 2 lists the best system, our system, and the baseline results for the NE-COARSE-LIT and EL-ONLY tasks.[11] Our NER system performs worse than the L3i system. However, we perform better than the provided baseline, which is a CRF sequence classifier. We have an exception to the French NE-COARSE-LIT-strict result, which is slightly below the baseline F1-score.

It turns out that our EL system, especially for English, performs better than the L3i team and the aidalight-baseline, which corresponds to [12]. Our French EL system is better than the L3i EL system in terms of recall but rather appalling in terms of precision.

Table 3 and Table 4 present our English and French NER and EL performance on the HIPE test data with detailed information on false positive and false negative numbers. Strict NER, which is a more demanding task, performs worse than fuzzy NER. Looking further at each class, we highlight that there are considerably misclassified PROD entities and thus contribute to numerous false negatives.

Table 3. NERC-COARSE-LIT results on the HIPE test data.

Lang	Evaluation	Label	P	R	F1	TP	FP	FN
EN	NE-COARSE-LIT-micro-fuzzy	ALL	**0.568**	**0.746**	**0.645**	335	255	114
EN	NE-COARSE-LIT-micro-strict	ALL	0.461	0.606	0.524	272	318	177
FR	NE-COARSE-LIT-micro-fuzzy	ALL	**0.755**	**0.842**	**0.796**	1347	438	253
FR	NE-COARSE-LIT-micro-strict	ALL	0.605	0.675	0.638	1080	705	520

Table 4. EL-ONLY-LIT results.

Lang	Evaluation	Label	P	R	F1	TP	FP	FN
EN	NEL-LIT-micro-fuzzy-@1	ALL	0.633	0.685	0.658	305	177	140
EN	NEL-LIT-micro-fuzzy-relaxed-@1	ALL	0.633	0.685	0.658	305	177	140
EN	NEL-LIT-micro-fuzzy-relaxed-@3	ALL	0.633	0.685	0.65	305	177	140
EN	NEL-LIT-micro-fuzzy-relaxed-@5	ALL	0.633	0.685	0.658	305	177	140
FR	NEL-LIT-micro-fuzzy-@1	ALL	0.585	0.65	0.616	1039	737	560
FR	NEL-LIT-micro-fuzzy-relaxed-@1	ALL	0.604	0.67	0.635	1072	704	527
FR	NEL-LIT-micro-fuzzy-relaxed-@3	ALL	0.604	0.67	0.635	1072	704	527
FR	NEL-LIT-micro-fuzzy-relaxed-@5	ALL	0.604	0.67	0.635	1072	704	527

6 Conclusion

As our participation in the HIPE shared task, we highlight that the quantity and quality of data need are essential for the NERC and the NEL tasks. Further, although

[11] https://github.com/impresso/CLEF-HIPE-2020/blob/master/evaluation-results/ranking_summary_final.md

DeLFT and *entity-fishing* achieve good F1-scores, their performance is quite sensitive to noise data.

Acknowledgements We thank the anonymous reviewers for their insightful comments.

References

1. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
2. Chiu, J.P., Nichols, E.: Named entity recognition with bidirectional lstm-cnns. Transactions of the Association for Computational Linguistics **4**, 357–370 (2016)
3. Delft. https://github.com/kermitt2/delft (2018–2020)
4. 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)
5. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: Overview of CLEF HIPE 2020: Named Entity Recognition and Linking on Historical Newspapers. In: Arampatzis, A., Kanoulas, E., Tsikrika, T., Vrochidis, S., Joho, H., Lioma, C., Eickhoff, C., Névéol, A., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the 11th International Conference of the CLEF Association (CLEF 2020). Lecture Notes in Computer Science (LNCS), vol. 12260. Springer (2020)
6. Foppiano, L., Romary, L.: entity-fishing: a DARIAH entity recognition and disambiguation service. In: Digital Scholarship in the Humanities . Tokyo, Japan (Sep 2018), https://hal.inria.fr/hal-01812100
7. Habibi, M., Weber, L., Neves, M., Wiegandt, D.L., Leser, U.: Deep learning with word embeddings improves biomedical named entity recognition. Bioinformatics **33**(14), i37–i48 (2017)
8. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016)
9. Luo, G., Huang, X., Lin, C.Y., Nie, Z.: Joint entity recognition and disambiguation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 879–888 (2015)
10. Ma, X., Hovy, E.: End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354 (2016)
11. Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A.: Advances in pre-training distributed word representations. arXiv preprint arXiv:1712.09405 (2017)
12. Nguyen, D.B., Hoffart, J., Theobald, M., Weikum, G.: Aida-light: High-throughput named-entity disambiguation. LDOW **1184** (2014)
13. Passos, A., Kumar, V., McCallum, A.: Lexicon infused phrase embeddings for named entity resolution. arXiv preprint arXiv:1404.5367 (2014)
14. 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)
15. Peters, M.E., Ammar, W., Bhagavatula, C., Power, R.: Semi-supervised sequence tagging with bidirectional language models. arXiv preprint arXiv:1705.00108 (2017)

16. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. arXiv preprint arXiv:1802.05365 (2018)
17. Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., Zhang, Y.: Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In: Joint Conference on EMNLP and CoNLL-Shared Task. pp. 1–40 (2012)
18. Ratinov, L., Roth, D.: Design challenges and misconceptions in named entity recognition. In: Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009). pp. 147–155 (2009)
19. Sagot, B., Richard, M., Stern, R.: Annotation référentielle du Corpus Arboré de Paris 7 en entités nommées. In: Antoniadis, G., Blanchon, H., Sérasset, G. (eds.) Traitement Automatique des Langues Naturelles (TALN). Actes de la conférence conjointe JEP-TALN-RECITAL 2012, vol. 2 - TALN. Grenoble, France (Jun 2012), https://hal.inria.fr/hal-00703108
20. Sang, E.F., De Meulder, F.: Introduction to the conll-2003 shared task: Language-independent named entity recognition. arXiv preprint cs/0306050 (2003)
21. Santos, C.N.d., Guimaraes, V.: Boosting named entity recognition with neural character embeddings. arXiv preprint arXiv:1505.05008 (2015)