

# SemEval-2020 Task 6: Definition extraction from free text with the DEFT corpus

Sasha Spala<sup>1</sup>, Nicholas A Miller<sup>1</sup>, Franck Deroncourt<sup>2</sup>, Carl Dockhorn<sup>1</sup>

<sup>1</sup>Adobe Inc., <sup>2</sup>Adobe Research

345 Park Ave

San Jose, CA 95110-2704

{sspala, nimiller, deronco, cdockhorn}@adobe.com

## Abstract

Research on definition extraction has been conducted for well over a decade, largely with significant constraints on the type of definitions considered. In this work, we present DeftEval, a SemEval shared task in which participants must extract definitions from free text using a term-definition pair corpus that reflects the complex reality of definitions in natural language. Definitions and glosses in free text often appear without explicit indicators, across sentences boundaries, or in an otherwise complex linguistic manner. DeftEval involved 3 distinct subtasks: 1) Sentence classification, 2) sequence labeling, and 3) relation extraction.

## 1 Introduction

Definition extraction as a complex, real-world task is currently an emerging field of study. Traditional definition extraction approaches mostly rely on simple, syntactically straight-forward examples with relatively little variance in vocabulary. Corpora, including the WCL (Navigli et al., 2010) and ukWaC (Ferraresi et al., 2008), typically consist of “definition sentences” which follow a standard  $X$  is a (type)  $Y$  or  $X$ , such as  $Y$  syntactic structure. Many also contain definitors (Navigli and Velardi, 2010), such as “means”, “is”, or “is defined by”. We provide a complete review of the existing state of definition extraction in Spala et al. (2019).

## 2 Data

The DeftEval task provided data from the DEFT corpus for training, development and testing. Introduced in Spala et al (2019), the DEFT corpus is currently the largest and most comprehensive corpus explicitly for definition extraction. In addition to the typical definition-type sentences discussed in the introduction, the corpus also contains sentence and “sentence windows” of wide variance in syntactic and semantic construction. Sentences for the corpus were retrieved from the open source textbook website cnx.org, as well as from various 2017 SEC contract filings from the publicly available US Securities and Exchange Commission EDGAR (SEC) database. For the DeftEval shared task, participants were only required to use sentences extracted from textbooks. EDGAR annotations were provided on request, but not required for participation in the shared task.

The DEFT corpus provides sentence “context windows” for textbook data from cnx.org to narrow the search space for possible definitions. A context window is defined as a set of three sentences: one sentence before a sentence containing a potential term (e.g. a bold word), the main sentence with the potential term, and one sentence after. Though it is certainly possible that some definitions may occur further away from the bolded mention of the term, for the purposes of the shared task, definitions that occur outside of the three-sentence window around a potential term are out of scope.

The corpus contains 9 different token labels (see Table1), and 5 different types of relations (see Table2). This expanded annotation schema allows for a wide range of data collection, as well as a better understanding of “non-traditional” definition types (i.e. beyond the typical  $x$  is a  $y$  structure). With this

---

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: <http://creativecommons.org/licenses/by/4.0/>.

Tag Name	Description
Term	A primary term
Alias Term	A secondary, less common name for the primary term. Links to a term tag.
Ordered Term	Multiple terms that have matching sets of definitions which cannot be separated from each other without creating a non-contiguous sequence of tokens. E.g. <i>x and y represent positive and negative versions of the definition, respectively</i>
Referential Term	An NP reference to a previously mentioned term tag. Typically <i>this/that/these + NP</i>
Definition	A primary definition of a term. May not exist without a matching term.
Secondary Definition	Supplemental information that may qualify as a definition sentence or phrase, but crosses a sentence boundary.
Ordered Definition	Multiple definitions that have matching sets of terms which cannot be separated from each other. See Ordered Term.
Referential Definition	NP reference to a previously mentioned definition tag. See Referential Term.
Qualifier	A specific date, location, or condition under which the definition holds

Table 1: Tag schema

Relation Name	Description
Direct-defines	Links definition to term.
Indirect-defines	Links definition to referential term or term to referential definition.
Refers-to	Links referential term to term or referential definition to definition.
AKA	Links alias term to term.
Supplements	Links secondary definition to definition.

Table 2: Relation schema

annotation, it is possible to observe long-distance (e.g. cross-sentence) term-definition relationships, as well as supplemental definition content. *Term* and *definition* tags in the schema are treated largely similar to Storrer and Wellinghoff’s *definiens* and *definitior* tags, respectively (2006). A term is defined as a primary term in a sentence which is accompanied by a definition phrase or sentence within the sentence context window. Terms are most commonly expressed as noun phrases:

*The scientific method* is a method of research with defined steps that include experiments and careful observation.<sup>1</sup>

Occasionally, terms may appear as verb phrases or other constituent phrases (e.g. adjective phrases):

Many crystals and solutions rotate the plane of polarization of light passing through them. Such substances are said to be *optically active*.<sup>2</sup>

Not including minor tokenization and sentence break fixes made during the shared task, there were no major modifications to the dataset before use in the DeftEval task. Complete details on the dataset is discussed in Spala et al. (2019).

### 3 DeftEval Task

The DeftEval shared task focused on fostering an expanded understanding of definition extraction for complex data, while keeping results comparable to traditional definition extraction tasks. The shared task,

<sup>1</sup>[https://github.com/adobe-research/deft\\_corpus/blob/master/data/deft\\_files/dev/t1\\_biology\\_0\\_0.deft](https://github.com/adobe-research/deft_corpus/blob/master/data/deft_files/dev/t1_biology_0_0.deft)

<sup>2</sup>[https://github.com/adobe-research/deft\\_corpus/blob/master/data/deft\\_files/dev/t3\\_physics\\_1\\_101.deft](https://github.com/adobe-research/deft_corpus/blob/master/data/deft_files/dev/t3_physics_1_101.deft)

which ran from 4 Sept 2019 to 13 March 2020, was comprised of three subtasks: sentence classification, sequence labeling, and relation extraction.

Participants were provided data in accordance with the DeftEval timeline:

- **15 Aug 2019:** Practice data released
- **4 Sept 2019:** Training data released
- **19 Feb 2020:** Beginning of Subtask 1 and 2 evaluation period, unlabeled test data for Subtask 1 and 2 released
- **1 March 2020:** Beginning of Subtask 3 evaluation period, unlabeled test data for Subtask 3 released
- **13 March 2020:** End of competition period, labeled test data for all subtasks released

DeftEval was administered using Codalab Competitions, a free, open-source platform for machine learning competitions and data challenges. Instructions for participation, including details about the annotation schema and information about the evaluation process, were made available during the practice and training periods of the shared task. All data and scripts were disseminated through the DEFT Github repository<sup>3</sup>, and linked to on the Codalab landing page<sup>4</sup>. Inconsistencies in the data, bug fixes, and updates to tokenization errors were made regularly throughout the training period as participants surfaced questions and potential problems with the dataset.

### 3.1 Subtask 1: Sentence classification

In keeping with previous definition extraction tasks, we believed it to be important that the DeftEval shared task included a sentence classification subtask. Participants were provided a script<sup>5</sup> to convert the existing dataset in its sequence-labeling format to individual sentences with a binary label indicating whether or not the sentence contained a definition. For the purposes of this subtask, only sentences which contained token sequences labeled as *definition* in the annotated data were considered “definition sentences”. Sentences containing only *secondary definitions* were given a 0 label. Unlabeled test data for the evaluation period was provided pre-converted in the sentence classification format; participants were only required to predict the label of each sentence at test time.

### 3.2 Subtask 2: Sequence labeling

Because of the nature of the expanded annotation in the DEFT corpus in comparison to previous definition extraction corpora, we believed it to be particularly important that the DeftEval shared task include a sequence labeling task. The DEFT corpus contains a large amount of auxiliary labels not previously seen in existing corpora, and the ability to reliably extract these labels could indicate a model with a broader capability to identify complex and lengthy definition relationships. Ultimately, we hope for models that could both extract *and* pair (as seen in Subtask 3.3) related term and definition labels together, but given the potential for these two tasks to be particularly difficult and the novel nature of the DEFT corpus, the tasks were split into two subtasks for DeftEval.

The default format of the DEFT corpus data is similar to the CoNLL 2003 format (Tjong Kim Sang and De Meulder, 2003) and formatted as BIO data (Ramshaw and Marcus, 1995). Therefore, participants did not need to convert any of the data for Subtask 2. At test time, participants were provided sentences where each line contained a token, source text file, lower char bound and upper char bound. Participants were required to predict only the appropriate BIO tag.

---

<sup>3</sup>[https://github.com/adobe-research/deft\\_corpus](https://github.com/adobe-research/deft_corpus)

<sup>4</sup><https://competitions.codalab.org/competitions/22759>

<sup>5</sup>[https://github.com/adobe-research/deft\\_corpus/blob/master/task1\\_converter.py](https://github.com/adobe-research/deft_corpus/blob/master/task1_converter.py)

### 3.3 Subtask 3: Relation Extraction

The final DeftEval subtask focused on extracting individual term-definition pairs, as well as matching auxiliary labels to their appropriate root. Without having a precedent for this particular task in the definition extraction domain, it was difficult to predict the level of difficulty for this task. Given the nature of existing relation extraction tasks, we expected this task to be particularly complex.

Participants were again able use the default format of the DEFT data for training. At test time, participants were provided sentences where each line contained a token, source text file, lower char bound, upper char bound, BIO tag, and tag ID. Participants were required to predict both the tag ID of the sister relation and the relationship tag.

### 3.4 Task Logistics

All participants were required to use the provided data for training, but were allowed to add additional resources to their training data (e.g. other definition extraction resources, such as WCL, or general purpose models such as BERT). Additional hand-labeled definition extraction data, proprietary datasets, or unpublished data not universally accessible was explicitly disallowed. Participants were required to use the DEFT data as intended, using dev and training data *only* for training and development. Though it was only made available during the evaluation periods, participants were also informed that test data was to be used solely for testing purposes.

During the training period, participants were allowed unlimited submissions to the Codalab practice and training phases. The practice phase evaluated against the sample "practice" data, consisting of three "dummy" files, one for each subtask. The training phase evaluated against the dev set data. Exact copies of these evaluation data were provided in the Github repository. Participants were able to submit as many subtasks together as desired, though the evaluation process required that *all* files in a particular subtask be present for evaluation. In other words, participants must submit all files in the subtask evaluation set in order to be evaluated.

Evaluation scripts were also made available on the Github repository so that participants had access to understand and test the evaluation methods. Participants were encouraged to test their models locally on their own machines, then compare those results to those provided by the Codalab portal to ensure the formatting of their submissions were consistent with the Codalab requirements. Participants were also provided access to sample submissions via the Github repository for both the practice and training phases.

Each evaluation period ran for 12 days in which participants were able to submit their results through the Codalab submission process. Because of the nature of the relation extraction task potentially providing data for the sequence labeling task, the evaluation periods for Subtask 1 and 2 ran concurrently for 12 days, followed by the 12 day evaluation period for Subtask 3. Each participant was allowed 5 submissions per subtask, primarily to allow for any complications in the submission process. The vast majority of participants who submitted to the evaluation period used all 5 submissions. We suspect that many participants used these 5 submission attempts strategically in order to achieve higher results on the test sets, which should be taken into account for future tasks.

## 4 Shared Task Results

Over the course of the shared task competition period, 279 submissions were made through the Codalab training phase, with many participants additionally developing and testing their systems locally. The evaluation periods brought 217 submissions for Subtask 1, 220 for Subtask 2, and 113 for Subtask 3. Results are based on the evaluation metrics described in Section 3.

Participants in DeftEval 2020 used a various array of methods for the three subtasks. For Subtasks 1 and 2 (see Tables 3 and 4), many participants opted simply to use a pre-trained language model, especially BERT, RoBERTa, and XLNet (see Xie et al. (2020), Avram et al. (2020), Singh et al. (2020), Jeawak et al. (2020), Ranasinghe et al. (2020), and Davletov et al. (2020)). Other implementations incorporated various LSTM layers in addition to the use of BERT or other transformer architecture (see Zhang and Ren (2020)).

### Subtask 1 Results

rank	username	score	rank	username	score
1	ewrfcas	0.8783	30	bigwhitepear	0.7807
2	shifeideshi	0.8768	31	zhaoxf4	0.7807
3	semeval6	0.8708	32	andrei.avram	0.7772
4	lichaoaly	0.8531	33	yjhappy	0.7758
5	davletov-aa	0.8444	34	m1er	0.7753
6	caspa	0.8304	35	Hasaki	0.7731
7	zhangleikid	0.8239	36	gulshan73	0.7727
8	liuyukid	0.8167	37	zhanglonghui	0.7695
9	zhangjie001	0.8105	38	zhang-nlp	0.7663
10	ashatilov	0.8105	39	nishnik	0.766
11	yejingyi	0.8104	40	kpriyanshu256	0.7593
12	unixlong	0.8077	41	mukesh_1696	0.7555
13	heyilong	0.8042	42	kuldeep7688	0.7555
14	neuysj	0.8028	43	m.sat1awa	0.7395
15	lll_2020	0.8008	44	KerenzaDoxolodeo	0.7391
16	Shsabab	0.8007	45	aadarshsingh191198	0.7317
17	caspiar	0.8	46	deftpunk	0.7109
18	zhanghh2020	0.7971	47	Haemant	0.6851
19	guo222	0.7914	48	mkannan	0.6801
20	happyyuja	0.7913	49	Zawawy	0.6757
21	S.L.Liu	0.7905	50	nibollocksnode	0.6741
22	Sattiy	0.7898	51	dpc	0.6466
23	juaeKim	0.7894	52	ruspajina	0.6035
24	yyyang	0.7894	53	junelady	0.4903
25	plumaj	0.7885	54	michael_wzhu91	0.4903
26	tharindu	0.7885	55	GIL-UNAM	0.4865
27	yunfeng4	0.7879	56	tangmen	0.2998
28	MagiJohnson	0.7858			
29	iman_saleh	0.7828			

Figure 1: Subtask 1 results; 56 participants, 217 total successful submissions

### Subtask 2 Results

rank	username	score	rank	username	score
1	ewrfcas	0.8471	27	marchbnr	0.4968
2	zhangdashuai	0.7859	28	plumaj	0.4872
3	yejingyi	0.7853	29	lll_2020	0.4737
4	bert_knows	0.7853	30	guo222	0.4715
5	kobe4ever	0.7813	31	yjhappy	0.4694
6	wamgnowem	0.7788	32	happyyuja	0.4665
7	zhangjie001	0.7749	33	iman_saleh	0.4651
8	Honda	0.7706	34	tharindu	0.4589
9	septboy	0.7559	35	ashatilov	0.4583
10	nlp_stuff	0.7372	36	Sattiy	0.4533
11	albert_knows	0.7329	37	andrei.avram	0.4398
12	junelady	0.727	38	furkancoskun	0.4356
13	michael_wzhu91	0.7112	39	giftai	0.4113
14	forever_kb	0.7101	40	m.sat1awa	0.3722
15	unixlong	0.6869	41	caspiar	0.3418
16	caspa	0.6765	42	yunfeng4	0.3234
17	ai_not_intellegent	0.6417	43	zhang-nlp	0.3234
18	lciy	0.5933	44	m1er	0.2899
19	juaeKim	0.5668	45	mukesh_1696	0.2852
20	shifeideshi	0.5306	46	kuldeep7688	0.2577
21	semeval6	0.5306	47	paradoxlover	0.1995
22	zhangleikid	0.5234	48	KerenzaDoxolodeo	0.1395
23	davletov-aa	0.5233	49	GIL-UNAM	0.0971
24	yyyang	0.5154	50	neuysj	0.0219
25	lichaoaly	0.5151	51	tangmen	0.0025
26	liuyukid	0.5107			

Figure 2: Subtask 2 results; 51 participants, 220 total successful submissions

Subtask 3 Results

rank	username	score	rank	username	score
1	unixlong	1.0	15	Cheoneum	0.9043
2	wamgnowem	0.9943	16	yyyang	0.9027
3	forever_kb	0.9943	17	jujbob	0.8866
4	caspa	0.9943	18	m.satlawa	0.8273
5	zhangdashuai	0.9933	19	yejingyi	0.7938
6	zhangjie001	0.9928	20	ai_not_intellegent	0.6508
7	ewrfcas	0.9924	21	davletov-aa	0.6097
8	nlp_stuff	0.9918	22	ashatilov	0.4874
9	Honda	0.9918	23	math11	0.4
10	septboy	0.9908	24	Ms8qQxMbnjMgYcw	0.3566
11	albert_knows	0.9715	25	GIL-UNAM	0.3367
12	Sattiy	0.9223	26	III_2020	0.2139
13	juaeKim	0.9049	27	andrei.avram	0.1754
14	king001	0.9043			

Figure 3: Subtask 3 results; 27 participants, 113 total successful submissions

Rank	Team Name	Submission Name	Reference	Models
5	*	davletov-aa	Davletov et al. (2020)	Multi-task BERT
6	ACNLP	caspa	Caspani et al. (2020)	RoBERTa + Stochastic Weight Averaging
12	UNIXLONG	unixlong	Xie et al. (2020)	Joint classification and sequence labeling pre-trained model with MLP and CRF layer
16	Cardiff University	Shsabab	Jeawak et al. (2020)	BERT with two-step fine tuning
18	BERTatDE	zhangh2020	Zhang and Ren (2020)	BERT with BiLSTM + attention
26	RGCL	tharindu	Ranasinghe et al. (2020)	XLNet
32	UPB	andrei.avram	Avram et al. (2020)	RoBERTa with finetuning
40	DSC IIT-ISM	kpriyanshu256	Singh et al. (2020)	BERT with fine-tuned language model
46	DeftPunk	deftpunk	Soboleva and Kaparina (2020)	FastText and ELMo embeddings with RNN ensemble
47	TüKaPo	haemanth	Kannan and Ponnusamy (2020)	Concatenated GloVe and on-the-fly POS embeddings with BiLSTM and 1D-Conv + MaxPool layers

Table 3: Subtask 1 submissions and system descriptions.

\*Team name not specified

Rank	Team Name	Submission Name	Reference	Models
23	*	davletov-aa	Davletov et al. (2020)	BERT
27	defx	marchbnr	Hübner et al. (2020)	CRF tagger
34	RGCL	tharindu	Ranasinghe et al. (2020)	XLNet - large
37	UPB	andrei.avram	Avram et al. (2020)	RoBERTa + CRF with fine-tuning

Table 4: Subtask 2 submissions and system descriptions.

\*Team name not specified

Rank	Team Name	Submission Name	Reference	Models
1	UNIXLONG	unixlong	Xie et al. (2020)	BERT + hand crafted rules
4	ACNLP	caspa	Caspani et al. (2020)	Random Forest

Table 5: Subtask 3 submissions and system descriptions.

Participants were also particularly interested in multi-task solutions for Subtasks 1 and 2, to varying degrees of success (see Xie et al. (2020), Avram et al. (2020), ?). Avram et al. (2020) used pre-trained embeddings, then “projected the contextualized embeddings generated by the RoBERTa model in three vectors”, making predictions for all three subtasks with a cohesive architecture.

Some “classic” neural architectures were also used in participant submissions to Subtask 1, including Kannan and Ponnusamy (2020)’s LSTM, RNN, and “hybrid” CNN solution, which leverages “the implicit local feature-extraction performed by the convolutional layers to refine the final representations passed to the recurrent layer, which accounts for global features.” Ranasinghe et al. (2020) also used CNN and RNN methods on Subtask 1, in addition to other pretrained architectures.

Subtask 3 submissions (see Table 5) were generally more simple in nature, perhaps due to the issues discussed in Section 5.2. Xie et al. (2020) used a pre-trained BERT model with some rule-based layers, and Caspani et al. (2020) used a random forest approach with additional post-processing.

In addition to an array of architectures, DeftEval also sparked conversation around the annotation schema. Jeawak et al. (2020) point out that the complexity of the schema make this task, as expected, more difficult than previous definition extraction tasks. They also note that some annotations conflict with each other in the training and test sets, which may be due to the nature of the schema, as well as the complicated content of the documents annotated. We recognized this issue, and recommend that future definition extraction tasks and tasks using the DEFT data consider this. Others mentioned they found that their initial model explorations came up short; Singh et al. (2020) mention that “owing to the complexity of the DEFT corpus” their “syntactically aware neural networks” performed lower than expected.

Overall, participants employed a variety of different models, especially those based on pre-trained transformer architectures for Subtasks 1 and 2 and more “classic” approaches for Subtask 3. Participants also pointed out discrepancies and issues with the source dataset, which will be accounted for in future work.

## 5 Complications

### 5.1 Amendments to the DEFT Corpus

The source data for DEFT was extracted by an automatic script that collected and sentence tokenized content from cnx.org. While modern automatic tokenization methods generally work well in clean environments, content scraped from cnx.org often unintentionally includes raw urls, incorrectly parsed text, and errant html code. Because of this, the tokenization process for the textbook data was particularly er-

ror prone. Though our annotators and developers spent significant time manually adjusting tokenization errors, many still existed at the time of release on Github. Many participants discovered these tokenization errors during training, which were then verified and fixed via Github issues. Because of the mixed nature of the source problem for tokenization errors, many caused incorrect sentence tokenization, which in turn meant that the actual number of sentences in the corpus was different from the reported number in (Spala et al., 2019). With the existing tokenization fixes at publication time, the textbook corpus contains 16056 sentences (695627 tokens) and 9439 2-3 sentence context windows.

## 5.2 Subtask 3 difficulty

In organizing Subtask 3, we believed the task of matching term-definition pairs to be particularly difficult, especially considering the number of additional possible tags in the DEFT corpus. Given the precedence of previous performance on other more constrained relation extraction tasks (see also Lin et al. (2016), Shen and Huang (2016), Sorokin and Gurevych (2017), etc.), we predicted this task would see mostly low performing scores.

However, participants in the subtask received a median F1 score of 0.9043, with 16 of the 27 participants scoring above 0.9. Upon further inspection, we realize this is due to the narrow search space of the three-sentence context windows. In designing the subtasks, we decided to keep the context windows in the test set so that the format of the data remained the same between released training data and test data and to minimize confusion for an already complicated task. Unfortunately, most context windows only include one set of term, definition, and matching auxiliary tags, meaning that this task was in fact particularly easy. F1 scores of 1 and near 1 make sense given the fact that models had to predict scenarios which in many cases only had one possible answer. Only cases that had more than one term-definition pair had a search space larger than 1.

We recommend that future iterations of this particular task provide sentences in context with the entire document. Though this does not work with the current format of the DEFT corpus, providing annotated data in context with the rest of the source textbook is a trivial task. While context windows only show the range of three sentences and possible distance between term-definition pairs would remain within those three sentences, providing the full context of the textbook would mean that models would have to discern term-definition pairs that cross context windows or appear in neighboring sentences. Future tasks may also consider combining Subtasks 2 and 3 in order to create a more comprehensive and challenging task overall.

## 5.3 Concurrent subtask evaluation periods

The evaluation periods for Subtasks 1 and 2 were scheduled to run concurrently in order to maximize the amount of time allotted for each subtask without extending the official SemEval evaluation period and to account for the data needs of Subtask 3. While Codalab’s interface allowed for scheduling of two separate phases which had concurrent dates, the implementation in fact only allows for one phase. As a solution, we merged the two subtask evaluation phases into one phase which accepted both subtasks’ submissions. Unfortunately, this also meant that individuals were now only provided 5 total submissions for *both* subtasks. To handle this, participants were instructed to complete submissions for their Subtask 1 evaluation first, then contact the task organizers to open additional submissions for Subtask 2 evaluation, keeping their best scoring Subtask 1 results in their submission files so that both Subtask 1 and Subtask 2 scores would be available on the leaderboard. Participants were warned that if their submissions for Subtask 1 changed with their Subtask 2 data after their 5 allotted submissions, the results would be removed and they would be subject to removal from the leaderboard.

We recommend future tasks with multiple subtasks using Codalab to either run these subtasks on separate, non-overlapping dates, or to provide the total number of submissions needed for *both* subtasks in the phase instructions, and remove any submissions that exceed the allotted submissions for the individual subtasks.



## 6 Conclusion

Overall, we believe the DeftEval shared task was a successful first exploration of definition extraction with the DEFT corpus. With more than 400 submissions and 149 registered participants, definition extraction as a complex, real-world task is clearly of interest to many. We believe that the results from this shared task provide a baseline for expectations for more complicated definition extraction tasks.

Though this task does not test a production-ready or real-life scenario, where models may have to predict term-definition pairs in a large document or in a wider context, the task does provide an initial look at how models perform on broad, non-standard definition extraction annotations. We recommend that future iterations of this task pay close attention to the effect of the DEFT corpus' three sentence context windows at test time, especially as it relates to the differences between an artificially narrow search space and documents "in the wild". We also recommend that future research includes a critical examination of conflicting annotations in the DEFT corpus.

Overall, we believe DeftEval has surfaced questions and research directions relevant to expanding definition extraction into more complicated linguistic environments, where data is particularly messy and the natural language it stems from is particularly complex. DeftEval provided a launching point for the exploration of this type of problem in definition extraction, proved the field interest for such a task, and laid groundwork for future iterations of definition extraction tasks.

## References

- Andrei-Marius Avram, Dumitru-Clementin Cercel, and Costin Chiru. 2020. Upb at semeval-2020 task 6: Pre-trained language models for definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Fabien Caspani, Pirashanth Ratnamogan, Mathis Linger, and Mhamed Hajaiej. 2020. Cnlp at semeval 2020 task 6: A supervised approach for definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Adis Davletov, Alexander Shatilov, Nikolay Arefyev, Denis Gordeev, and Alexey Rey. 2020. Multi-task learning for definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Adriano Ferraresi, Eros Zanchetta, Silvia Bernardini, and Marco Baroni. 2008. Introducing and evaluating ukwac, a very large web-derived corpus of english. 01.
- Marc Hübner, Christoph Alt, Robert Schwarzenberg, and Leonhard Hennig. 2020. marchbnr at semeval-2020 task 6: Joint extraction of concepts and relations for definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Shelan Jeawak, Luis Espinosa-Anke, and Steven Schockaert. 2020. Cardiff university at semeval-2020 task 6: Fine-tuning bert for domain-specific definition classification. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Madeeswaran Kannan and Haemant Santhi Ponnusamy. 2020. Ui at sem-eval 2020 task 6 : Definition mining. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133.
- Roberto Navigli and Paola Velardi. 2010. Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1318–1327, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Roberto Navigli, Paola Velardi, and Juana Ruiz-Martínez. 2010. An annotated dataset for extracting definitions and hypernyms from the web. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010*, 01.
- Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*.

- Tharindu Ranasinghe, Alistair Plum, Constantin Orasan, and Ruslan Mitkov. 2020. Rgcl at semeval-2020 task 6: Neural approaches to definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Yatian Shen and Xuan-Jing Huang. 2016. Attention-based convolutional neural network for semantic relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2526–2536.
- Aadarsh Singh, Priyanshu Kumar, and Aman Sinha. 2020. Dsc iit-ism at semeval-2020 task 6: Boosting bert with dependencies for definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Anna Soboleva and Jekaterina Kaparina. 2020. Deftpunk at semeval-2020 task 6: using rnn-ensemble for the sentence classification. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Daniil Sorokin and Iryna Gurevych. 2017. Context-aware representations for knowledge base relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1784–1789, Copenhagen, Denmark, September. Association for Computational Linguistics.
- Sasha Spala, Nicholas A. Miller, Yiming Yang, Franck Dernoncourt, and Carl Dockhorn. 2019. DEFT: A corpus for definition extraction in free- and semi-structured text. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 124–131, Florence, Italy, August. Association for Computational Linguistics.
- Angelika Storrer and Sandra Wellinghoff. 2006. Automated detection and annotation of term definitions in german text corpora. In *Proceedings of the Conference on Language Resources and Evaluation (LREC)*.
- 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 - Volume 4, CONLL '03*, page 142–147, USA. Association for Computational Linguistics.
- ShuYi Xie, Jian Ma, Haiqin Yang, Jiang Lianxin, Mo Yang, and Jianping Shen. 2020. Unixlong at semeval-2020 task6: A joint model for definition extraction. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.
- Huihui Zhang and Feilang Ren. 2020. Bertatde at semeval-2020 task 6: Extracting term-definition pairs in free text using pre-trained model. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, December.