

# **\*\*ML Model and Dataset Mentions Annotation Guideline**

## **Abstract**

This guideline is the guideline for our joint ML model and dataset mention annotation task on CS publications. We provide term definitions of entities, which should guide the annotators to understand the task and do the annotation work. Considering the difficulty of the annotation task, we also provide examples from annotated publications, accompanying the term definitions. We would also like to emphasize that since this is the first version, the annotator should first read the whole guideline for understanding our task, and report problems when annotating the trial publication(s) back to us when using the guideline.

## **Resources**

This guideline is inspired by,

- SciERC annotation guideline, [http://nlp.cs.washington.edu/sciIE/annotation\\_guideline.pdf](http://nlp.cs.washington.edu/sciIE/annotation_guideline.pdf)
- ACL TEC annotation guideline, [https://www.researchgate.net/publication/283354086\\_The\\_ACL\\_RD-TEC\\_Annotation\\_Guideline\\_A\\_Reference\\_Dataset\\_for\\_the\\_Evaluation\\_of\\_Automatic\\_Term\\_Recognition\\_and\\_Classification?channel=doi&linkId=56372c6a08aebc004000dd8b&showFulltext=true](https://www.researchgate.net/publication/283354086_The_ACL_RD-TEC_Annotation_Guideline_A_Reference_Dataset_for_the_Evaluation_of_Automatic_Term_Recognition_and_Classification?channel=doi&linkId=56372c6a08aebc004000dd8b&showFulltext=true)

## **Annotation Tool**

INCEpTION 20.4.

## **Preliminary remarks**

Generics and coreferences.

Generic mentions refer to string spans that cannot be identified as one of the name-based entities [listed below](#). However, they are interesting in that they refer to *informal* mentions of ML models and datasets, and thus, should be captured. "Our dataset" (class1), "the model" (class1), "30 different existing computer vision datasets" (class2), "pretrained multilingual language models" (class2) and "both methods" (class3) are examples of generic mentions.

Some of the generic mentions, like "our dataset" or "the model", can have coreferences to name-based entities that have been annotated previously in the text. If possible, they should be linked to the last or next mention, and should be related as such using the "coreference" relation.

Determiners and articles.

See also ACL RD TEC Guideline, 3.3 "Determiners".

In general, annotators are instructed to not include articles like "a", "the". However, articles should be considered for *generic mentions only*. For example, the determiner "a" in the string span "a large-scale, diverse dataset" should be included. In some cases, the article needs to be included to guard the semantical meaning of the entity of interest.

Abbreviations.

See ACL RD TEC Guideline, 3.3 "Abbreviations".

Term-abbreviation sequence.

See ACL RD TEC Guideline, 3.3 "Term-abbreviation sequence".

Term broken by abbreviations.

See ACL RD TEC Guideline, 3.3 "Term broken by abbreviations".

Terms broken by modifiers.

As a general rule, if noun-based string spans are broken by one or more modifiers, annotate the whole string span. For example, in the string span "a large-scale, diverse dataset", "diverse" and "large-scale" are modifiers, and thus, the whole string span should be annotated.

Generic nouns.

Terms can be accompanied by generic nouns, e.g., the term "dataset" in the string span "the SQuAD v1.1 dataset" or "a large-scale, diverse dataset". We will consider these generic nouns for *generic mentions only* and not for named entities. Thus, "dataset" should not be included in the first example, but the second instead.

Proper nouns.

Proper nouns (names) are string spans corresponding to an explicit name-based mention in the text that relates to one entity of interest (see [Entity Category](#)). Examples are "Social Bias Inference Corpus (SBIC)", "BERT", "SQuAD", "GLUE". As stated earlier, [determiners and articles](#) should not be included during annotation, neither should [generic nouns](#).

Adjectival modifiers.

See also ACL RD TEC Guideline, 3.3 "Adjectival modifiers". In many cases, entities are modified by adjectives, for example, in the strings "well-known CommonCrawl" or "multilingual BERT (mBERT)". In these cases you should check whether the meaning is changed when the adjectives are removed. If yes, do not remove the adjective. For example, "multilingual" is an important modifier in the second example, whereas "well-known" is not important.

Adjectival modifiers should be included for generic mentions, e.g. "improved QA-pair models".

Conjunctions, prepositions, and “constructed entities”.

See ACL RD TEC Guideline, 3.3 "Conjunctions and prepositions".

Plurals.

We do not consider plural mentions like “the image datasets” as noun-based entities, in particular instances of a model or dataset. Instead, we treat them as generic mentions.

Scope of relations.

We limit the scope of relations to one paragraph.

Excluded from annotation.

Annotators are instructed to *not include* the “References” section. When annotators detect non-sentence format of paragraphs (e.x. Extraction of a table from PDF), then annotators should annotate the whole paragraph as Corrupted.

Other general remarks.

- Keep the annotations flat, do not stack much.
- Watch out for judgemental statements in descriptions.

## Term Definitions and Examples

### Entity Category

In this section, we provide the definition and corresponding annotation example for each entity. We note that each entity name is in font **Bold**, and the definition or description of the entity follows the entity names. Examples of annotation are screenshots from Inception tool.

#### *MLModel*

This entity type refers to a string span that represents a name based entity of a machine learning model. A MLModel usually is based on some machine learning (ML) architecture, and can be applied to some ML tasks. For neural network based machine learning models, such a string span should correspond to the executable resource of the model. We give an example in Figure 1.

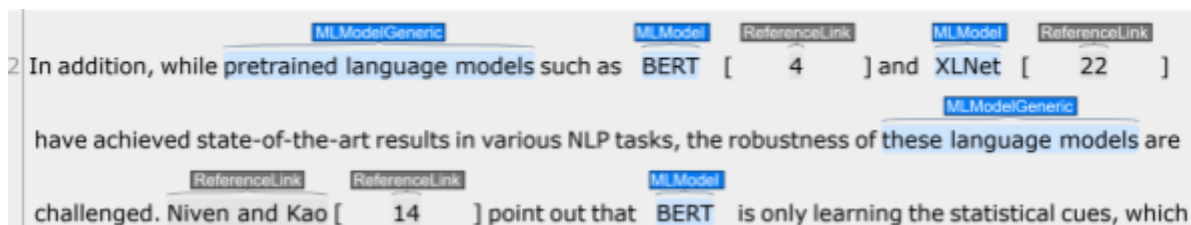


Figure 1. In this case, “BERT”, “XLNet” are MLModel instances.

### MLModelGeneric

This entity type refers to a string span corresponding to an MLModel in the text, which cannot be identified as a named entity. Typical examples are: “our model”, “the model” or plural mentions like “conventional models”, or “QA-pair retrievers”. For MLModelGeneric. Examples are given in Figure 2 & 3.

are not robust in both whitebox and blackbox settings. While we observe the impressive improvements using the pretrained language models, we expect our study can encourage further research into the robustness problems of current pretrained language understanding models. Attack Strategy. As we have

Figure 2. In this example, both “the pretrained language models” and “current pretrained language understanding models” are informal mentions of a group of machine learning models, thus are categorized as MLModelGeneric.

generate the output autoregressively. Taking TableQA as an example, given an NL question, our method generates the answer by decoding it in a word-by-word fashion.

In this paper, we present TAPEX, an execution-centric table pre-training approach whose corpus

Figure 3. We give two more examples of MLModelGeneric. “our method” is an anonymous mention to the machine learning model proposed by the authors, thus is categorized as MLModelGeneric. Note: this instance does not contain other information apart from the possessional relation between the referred model and the paper. “an execution-centric table pre-training approach” is also an anonymous mention of the model TAPEX, but it provides certain information regarding the referred model.

### ModelArchitecture

This entity type refers to a name based string span that mentions a ML model architecture or conceptual type. We clarify the difference between ModelArchitecture and other model related entity types as below:

- The difference between Method and ModelArchitecture is that a Method may contain other algorithmic steps and multiple model architectures.
- Certain name based spans can be assigned to either MLModel or ModelArchitecture according to the context. We assign such a span to ModelArchitecture when the semantics refer to the conceptual structure, or more specifically this structure is being used or based on or modified in the context.
- Between MLModelGeneric and ModelArchitecture, when the MLModelGeneric refers to a general type of some machine learning models, the string span should be double annotated as ModelArchitecture and MLModelArchitecture.

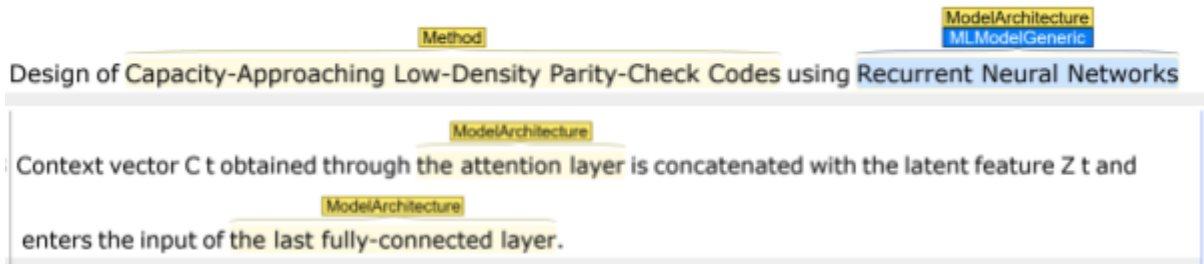


Figure 4. We give examples for ModelArchitecture in above screenshots.

### Method

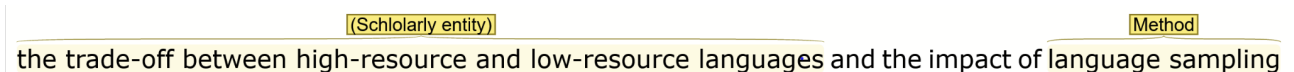
This entity type refers to a string span that refers to a *method*. Methods could be

- noun-based phrases from which a ModelArchitecture can be *derived*;
- *results* from a conceptualization, compared to an executable model with determined parameter values;
- things that are *produced by* a model, e.g. "GloVe embeddings", "ELMo embeddings", "word embeddings", "sentence embeddings";
- *components* of a model architecture that cannot be categorized as ModelArchitecture, e.g. "loss function", "Adam optimizer".

Methods spans should be considered as generics, and hence all rules apply.

### Note

1. Machine learning representations should be annotated as Method entities.
2. We should annotate "unsupervised learning", "supervised learning" etc. as Methods.
3. In the case of "GloVe embeddings", "ELMo embeddings", "GloVe" and "ELMo" should be stack annotated as ModelArchitecture.



### Dataset

This entity type refers to a string span corresponding to an explicit noun-based dataset object in the text (e.g. 'Social Bias Inference Corpus', "SBIC", 'SQuAD', 'GLUE') . If the string span includes both the full-length name and the acronym, we include them both separately as a Dataset (see [Preliminary Remarks "Abbreviations"](#)). An example is given in Figure 5.

**Note** A Dataset can contain a set of other Dataset instances, as in the case of "CommonCrawls" datasets. And the relation "isPartOf" can be used for these Dataset instances for this purpose.

The proposed algorithm was evaluated on classification tasks for CIFAR-10 & CIFAR-100 with a varying label noise ratio from 0% to 80%. We show results both for a clean validation set and a noisy one. In both

Figure 5. Examples of Dataset spans.

### DatasetGeneric

This entity type refers to a string span corresponding to a mention referring to one or more concrete named entity based datasets in the text, but is not an explicit name-based dataset.

**Note** Data source name (e.g. Wikipedia) can be included in a DatasetGeneric mention (e.g. “the Wikipedia dataset”, “the Wikipedia datasets”) .

samples obtained by annealed Langevin sampling and single step denoising jump. We evaluated Preprint 50k sampled images after training on CIFAR-10 with two performance scores, Inception ( Salimans et al., 2016) and FID (Heusel et al., 2017). We achieved Inception Score of 8.31 and FID of 31.7,

Figure 6. “50k sampled images” is not a name-based dataset mention, but provides additional type and size information in the mention.

Seasonal Variable As the results of the time-series decomposition in Figure 2, entrant data has seasonality . Thus, we use seasonal dummy variables to utilize the seasonality of the data in the predictive model. Robust Learning under Label Noise with Iterative Noise-Filtering

water quality are conducted. Measurements from 9 locations on the west side of the city are considered for analysis with mentioned locations spread over roughly 2 km length of the coastline. Locations of the

Figure 7. More DatasetGeneric examples.

### DataSource

This entity type refers to a string span of *data source information*, e.g. “Wikipedia”, “Twitter” and other possible explicit web or unstable, or not static sources. The difference to a Dataset is that the DataSource has unstable reproducibility.

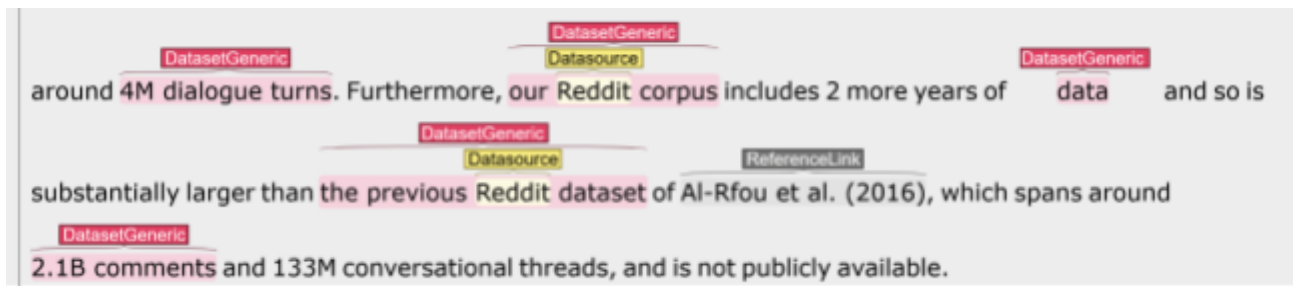
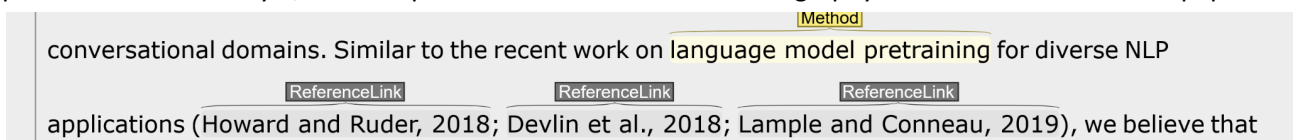


Figure 8. "Reddit" is a DataSource span.

### ReferenceLink

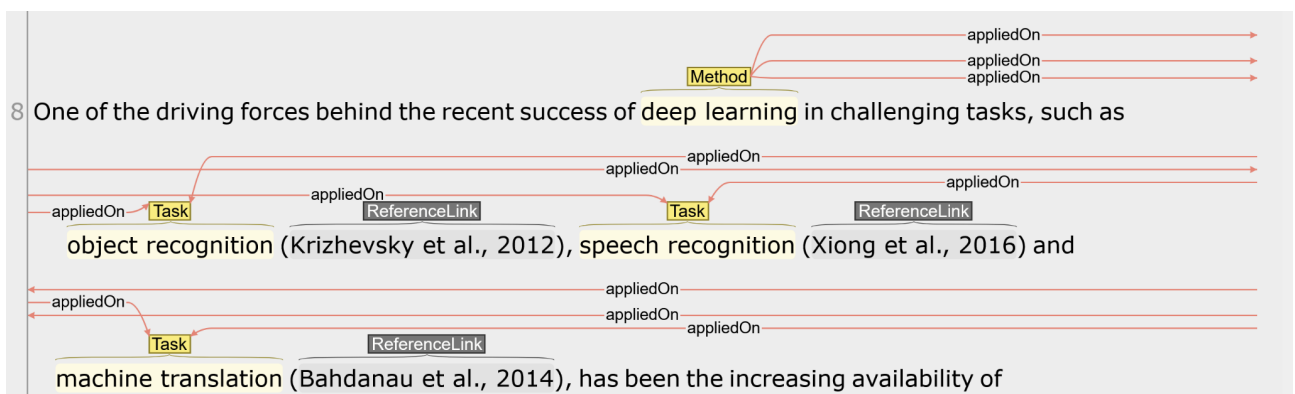
A ReferenceLink instance refers to a string span that represents a reference in the text. A ReferenceLink may present in different style, but it requires to be linkable to the bibliography section at the end of the paper.



### Task

This entity type refers to a string span which represents a specific ML task or a collection of themed ML tasks.

**Note** We only annotate a name-based string as a Task, and omit descriptive processes for a task. A counterexample is "removing the next sentence prediction objective".



### URL

An URL instance is a string span that is a url in the text.