RESEARCH PAPER

Automatic Extraction of Materials and Properties from Superconductors Scientific Literature

Luca Foppiano[a], Pedro Baptista de Castro[b], Pedro Ortiz Suarez[c], Kensei Terashima[b], Yoshihiko Takano[b], Masashi Ishii[a]

[a]Material Database Group, MaDIS, NIMS, Tsukuba, JP; [b]Nano Frontier Superconducting Materials Group, MANA, NIMS, Tsukuba, JP; [c]Data and Web Science Group, University of Mannheim, Mannheim, DE

ARTICLE HISTORY
Compiled August 1, 2022

ABSTRACT
The automatic extraction of materials and related properties from the scientific literature is getting more attention in data-driven materials science (Materials Informatics). In this paper, we discuss grobid-superconductors, our solution for automatically extract superconductor material names and respective properties from text. Built as Grobid module it combines Machine Learning and Heuristic approaches in a multi-step architecture supporting input data as raw text or PDF documents. Using grobid-superconductors, we built SuperCon^2, a database of 40324 materials-properties records from 37700 papers. The material (or sample) information is represented by name, chemical formula, material class, and is characterised by shape, doping, variables for components, substrate as adjoined information. The properties include the T_c superconducting critical temperature and, when available, applied pressure with T_c measurement method.

KEYWORDS
materials informatics, superconductors, machine learning, nlp, tdm

CLASSIFICATION: Machine learning, Text mining, NLP

1. Introduction

In recent years, with the creation of computational databases, such as the Materials Project (MP) [**?**], the Open Quantum Materials Database (OQMD) [**?**], and then experimental data repositories such as NIMS MDR (`http://mdr.nims.go.jp`) [**?**], the focus has been steadily shifting towards a data-driven design of materials which is often called as Materials Informatics (MI). Such an approach is expected to accelerate the exploration of functional materials, as it does not rely on the intuition of very little genius researchers nor on their limited experience. In this new paradigm, the efficient use of data to guide experiments and materials property prediction through the use of machine learning methods takes the centre stage. For example, data-driven methods have been used to search/design magneto-caloric materials [**?** **?** **?**], photo-catalysts for hydrogen splitting [**?**], thermoeletrics [**?**], and superconductors [**?**]. In such a

Corresponding authors: Luca Foppiano (luca@foppiano.org) and Masashi Ishii (ISHII.Masashi@nims.go.jp)

data-driven search, one of the most important keys lies in the availability of the data, that at least should consist of compositions of materials and their physical properties. In the specific case of superconductivity, most of the data-driven works [**?** **?** **?**] relies on a single database: SuperCon (`http://supercon.nims.go.jp`).

SuperCon is a structured database of superconductors materials and properties, developed at the National Institute for Materials Science (NIMS) in Japan. At the time of writing this paper, SuperCon contains about 33000 inorganic and 600 organic materials and is the "de-facto" standard in data-driven research for superconductors materials (about 4400 articles contain the mention "*supercon database*" in Google Scholar). However, SuperCon harvesting process is currently fully manual "from scratch": the humans have to read the human-readable printed matter such as PDF documents and enter the information in the system. The efficiency is directly proportional to the number of available human curators. Considering the cost of database construction, it is necessary to consider an assisted or alternative system that improves throughput while ensuring data quality equivalent to that of manual extraction.

As a solution, we are developing a hybrid data extraction methods from scientific literature combining automation using text data mining and manual curation. The automated system extracts and formats potential data and proposes them to the curator as "pre-cooked" structured data: (a) Highlight the relevant entities on the original document. (b) Pre-fill the extracted information in a tabular format. In building the automatic part of this hybrid system (training, evaluation) we used SuperMat [**?**], which we recently constructed.

In this work, we present *grobid-superconductors*: a system to extract automatically structured information of superconductors materials and properties from scientific literature. The tool is a specialised module of Grobid [**?**], a machine learning library designed to parse and structure scientific documents. Grobid provides an open source platform for building specialised modules: astronomical entities recognition[**?**], dictionaries [**?**], software mentions [**?**], and physical measurements extraction [**?**]. Grobid provides several built-in features including access to PDF document layout information, citation resolution, bibliographic information consolidation through *biblio-glutton* [**?**] a fast open-source reference matching service for CrossRef data, and a diverse set of ML architectures from a fast linear Conditional Random Field (CRF) to the latest state-of-the-art deep learning implementations.

Using grobid-superconductors and other sub-tools, we established a pipeline for processing a large number of documents and obtaining an automated database of superconductors materials and properties. We processed 37770 papers from ArXiv (`https://arxiv.org`) and obtained a database of 40324 records. This new database, named SuperCon^2, can become the automated staging area for SuperCon, bridged by a curation interface. The project was also an opportunity to focus on properties that gained interest in recent scientific trends and that are underrepresented in SuperCon. The "pressure" applied to obtain superconductivity (about 20 records in Supercon), has gained attention because it can change radically the physical structure of a material. The "method" used to measure the superconducting transition temperature T_c (about 600 records in SuperCon) can be used to semantically recognise multiple T_cs obtained from the same material or sample (e.g. distinguish calculated and experimental T_c).

2. Grobid-superconductors

Grobid-superconductor is a web application for processing text or PDF documents to extract materials and corresponding properties. We develop *grobid-superconductors* as a *Grobid* library [**?**] module following some principles (multi-step, sentence-based, fulltext-based) discussed in a previous preliminary study [**?**]. *Grobid* bring several advantages: a) It integrates with *pdfalto*[1], a specialised tool for converting PDF to XML which mitigates extraction issues such as the resolution of embedded fonts, invalid character encoding, and the reconstruction of the correct reading order. b) It allows access to PDF document layout information for both machine learning and document decoration (e.g. coordinates in the PDF document) and, finally, c) it provides access to a set of high-quality, pre-trained machine learning models for structuring documents. *Grobid-superconductors* is structured as a three-steps process as illustrated in Figure **??** and described in the following sections **??**, **??**, and **??**.

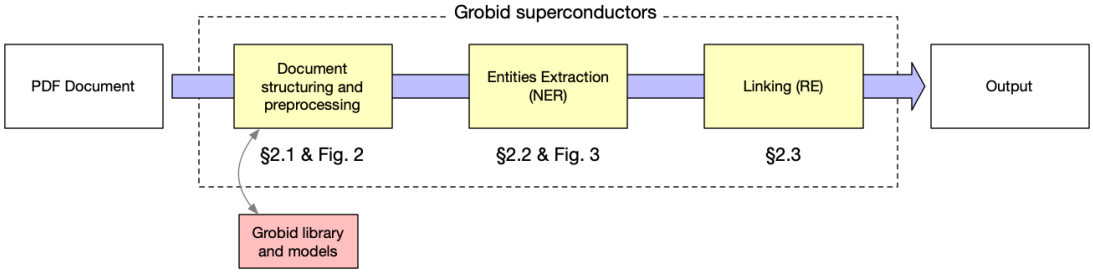

Figure 1. Processing pipeline for extracting superconductors materials and properties.

Abstract versus fulltext At the time of writing this paper, we are aware of related works that utilise text from abstracts as training data for machine learning. The main reason is that abstracts are usually freely available as text [**?**], and contain condensed information [**? ?**]. Accurately parsing the full text presents more challenges, however, they are mitigated by the *Grobid* library and the full texts contain a broader range of information, including the sample preparation process, negative results (e.g. absence of superconductivity for certain samples), and background information (e.g. report on other materials from referenced works). Thus, *grobid-superconductors* is built to support full-text documents.

Paragraphs versus sentences Another question related to NLP processing is whether to use sentence-based or paragraph-based text. While paragraphs can be extracted as part of the layout of PDF documents, obtaining sentences need to add an additional step which processes text using a sentence segmenter. However, sentences are smaller by definition (English writing guidelines recommend less than 25 words) and in deep learning, this brings some advantages. In training and prediction, sentences will likely be shorter than the "max sequence length" limitation (e.g. 512 tokens for transformers). In training sentences use less memory and can be trained with a larger "batch size", which provide better results [**?**].

We decided to use sentence-based text in *grobid-superconductors* after performing experiments on a smaller scale, applied to our tasks. For the NER tasks, we trained and

[1] `https://github.com/kermitt2/pdfalto`

evaluated a sequence labelling model for each version (paragraphs-based and sequence-based) on four annotated documents (3/1 documents partition for training/evaluation) from SuperMat [**?**]. As indicated in Table **??**, the F1-score increase by 17.94 points % by using sentence-based text.

Label	**Precision**	**Recall**	**F1**
Paragraph-based micro avg.	44.44	27.21	33.76
Sentence-based micro avg.	48.41	50.00	51.70

Table 1. Results from cross-validation using sentence-based or paragraphs based.

In the Linking task we want to maximise precision. In our previous work [**?**] we noticed that limiting linking entities within the same sentence (versus paragraphs) would obtain higher precision (68.7% versus 57%) at the expense of lower recall (6.5% versus 10.7%), and F1-score (11.87% versus 18.01%). In both our tasks we found evidences that a sentence-based dataset is more beneficial than paragraph-based dataset.

2.1. Document structuring and pre-processing

In the first step in our process, the PDF document is converted into an internal model based on a list of text statements, tokens, and features. The input document is processed using the Grobid original models where we apply customised processes for document header and content. We select a subset of bibliographic information from the header: title, authors, DOI, publisher, journal, and year of publication and we consolidate them via Grobid to match the publisher's quality (even by processing the "preprint version" of the publication). The superconductors entities extraction is applied to the content, only on relevant text items: title, abstract, text content from body or annexe, text content from figure and table captions (Figure **??**).

We use the collected reference markers (also called *reference callout*) from text as features for improving the paragraph segmentation in sentences: the segmentation is cancelled if the "end of sentence" falls within the boundaries of a reference marker. For example, a sentence containing a reference in the form "*Foppiano et. al.*" may be mistakenly segmented at "*et.*".

2.2. Entities Extraction

The second step is the *Entities Extraction* performing the Named Entity Recognition (NER) task on previously extracted text.

Overview

As illustrated in Figure **??** the "superconductors parser" extracts the main superconductors-related information by aggregating the resulting entities from two ML models. The *superconductors ML model*, developed based on the SuperMat schema [**?**], and the *quantities ML model*, developed in a separated grobid-module for measurement extraction [**?**] for which we limit the output to only temperatures and pressures. Overlapping entities are merged: exact duplicates are removed and the largest entities (in terms of string length) are preserved. The resulting entities are summarised in Table **??**.

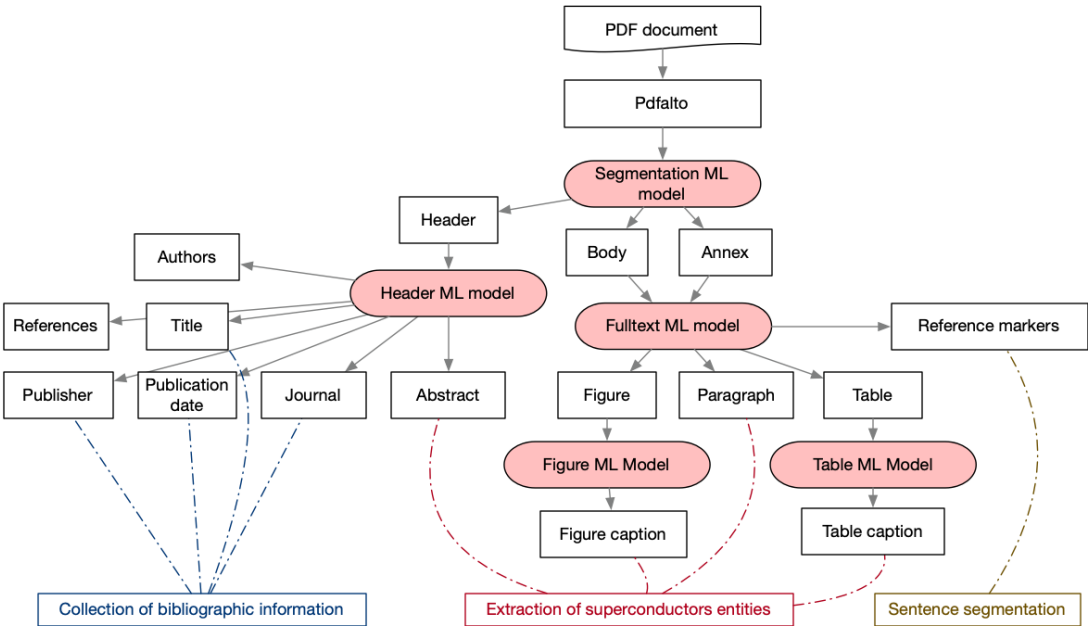

Figure 2. Grobid-superconductors extraction processes (bibliographic information, superconductors entities extraction and sentence segmentation) within the Grobid cascade data flow.

Entity (**Tag**)	**Description**
	Machine learning
Material (`<material>`)	Materials and samples names, formulas, including stochiometric formulas, substitution variables of values and elements, shape, doping, substrate
Class (`<class>`)	Groups of materials having similar characteristics or common strategic compounds that define their nature
Tc value (`<tcValue>`)	The value of the superconductors critical temperature
Tc expressions (`<tc>`)	Expressions in the text that provide information about the phenomenon of superconductivity related to a value, interval or variation of the T_c
Measurement methods (`<me_method>`)	Techniques used to measure or calculate the presence of superconductivity.
Applied pressure (`<pressure>`)	Applied pressure when superconductivity is recorded

Table 2. Synthesis of the superconductors parser entities.

The entities of type `<material>` are passed in cascade to the "material parser" which combines ML and tools to extract further information and structures. First, the material raw string is passed through a *material ML model* to segment the raw material string (Table **??**). Then, we apply different processes, based on which information is available:

- formulas are decomposed into a structured composition. We identify each pair of element-stoichiometry (e.g "O": 7.0) using mat2chem [**?**] and Pymatgen [**?**],
- if only the name is available, we obtain the formula (e.g. hydrogen to H),
- using heuristics, we classify the formula with tags that follows the superconductors researcher "material class" conventions, for example Cuprate, Oxides, Alloys, etc.
- using the variables and values extracted, we substitute them into partial formulas. For example, in `La 4 Fe 2 A 1-x O 7 (A=Mg,Co; x=0.1,0.2)`, we substitute A and x using their parsed values, and applying permutations we obtain four *resolved formulas*: `La 4 Fe 2 Mg 0.9 O 7`, `La 4 Fe 2 Mg 0.8 O 7`, `La 4`

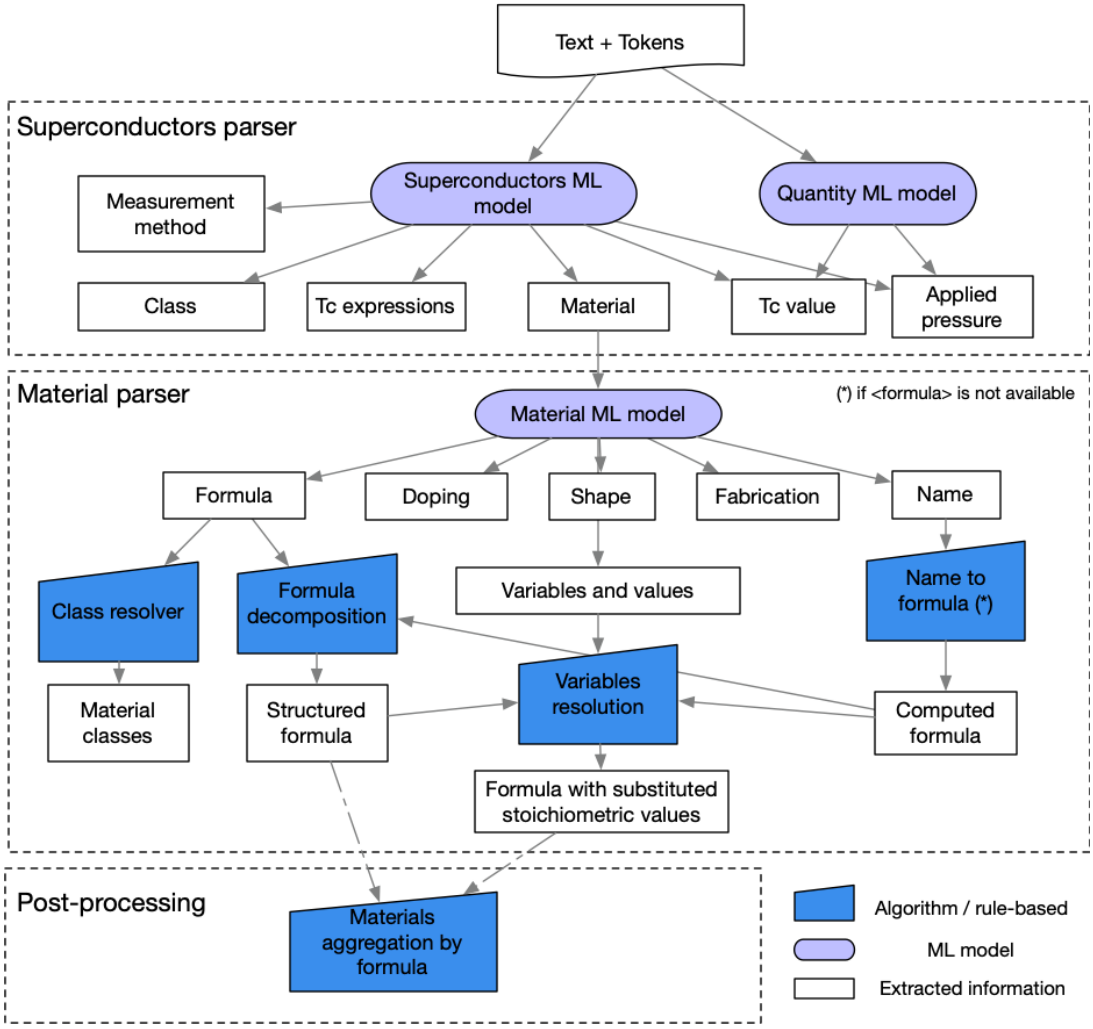

Figure 3. Cascade architecture in the Entities Extraction step. The white rectangles indicate the extracted information (described in Tables **??** and **??**). The ML models and the rule-based algorithms are identified by light grey and dark grey shapes, respectively.

`Fe 2 Co 0.9 O 7`, `La 4 Fe 2 Co 0.8 O 7`.

Finally, after all entities are extracted, the post-processing aggregates different mentions of the same materials using the parsed formulas at document-level. For example `hydrogen` and `H`, or formula with partial substitutions such as `La 2 Fe 1-x O 7 (x = 0.1, 0.2)` will be aggregated with materials like `La 2 Fe 0.9 O 7` appearing in other sections of the same document.

Machine Learning study

In this section we discuss the novel ML models we have trained for extracting specialised entities: *superconductors ML model* and *material ML model* (Figure **??**). SuperMat [**?**], our training dataset, counts 162 papers at the time of writing and is composed of annotated full-text and layout features from PDF documents.

For both ML models we trained and evaluated these four architecture/implementations:

Entity (Tag)	Description
Name (<name>)	The canonical name of a material (e.g. Hydrogen, PCCO, Carbon)
Formula (<formula>)	Chemical formula of the material (e.g. Pr1.869Ce0.131CuO 4-, MgB2, La 2-x Sr x CuO 4)
Doping (<doping>)	Doping ratio and doping materials that are adjoined to the material name (e.g. Zn-doped, 2% Zn-doped)
Shape (<shape>)	shape of the material (e.g. single crystal, polycrystalline, thin film, powder, film)
Substitution variables (<variable>)	Variables that can be substituted in the formula.
Substitution values (<value>)	Values expressed in the doping.
Substrate (<substrate>)	Substrates as defined in the material name
Fabrication (<fabrication>)	Represent eventual additional information that are not belonging to any of the previous tags (e.g. intercalated, electron-doped)

Table 3. Synthesis of the material parser entities.

- Linear CRF (CRF)
- Bidirectional LSTM with CRF [**?**] (BidLSTM_CRF)
- Bidirectional LSTM with CRF with Features [**?**] is the same as the previous one with an additional input channel for features (BidLSTM_CRF_FEATURES)
- SciBERT [**?**] using a CRF as activation layer (Scibert)

The ML models are interfaced by Grobid, which uses the Wapiti[**?**] implementation for linear CRF, and DeLFT (Deep Learning For Text) [**?**] for deep learning models. The architectures CRF and BidLSTM_CRF_FEATURES make use of orthogonal features we have summarised in Table **??**.

Superconductors ML model

Holdout set The holdout set must be balanced and should follow the same distribution of the target dataset. We assembled the holdout set by manually selecting from SuperMat 32 documents (24%) with the same ratio of training examples, entities and unique entities (Figure **??**a). Maintaining the same rate (80/20) for entity type distribution is more challenging: in average, we obtained around 15-18% of labels of each type in the holdout set (Figure **??**b), except for the <material> label (23%). The remaining 76% (132 documents) is used for training.

We define the "out-of-domain" ratio as the number of unique entities from the holdout set that are not in the training set. The holdout set "out-of-domain" ratio is on average around 72%, which requires the model to be able to generalise well. All the labels have "out-of-domain" ratio above 24% (Figure **??**) with the highest for the <material> at 82%. The labels <me_method> have the lowest "out-of-domain" ratio at 25% which can be explained by a high uniqueness of entities (355 entities are reduced to 57 unique).

Positive sampling We train the model with positive sampling by removing the examples without entities (negative examples, Figure **??**a). Compared with no sampling, this approach provides an improvement by +2% in both precision and recall, when testing against the holdout set. Further experiments with active and random sampling with 0.1, 0.25, 0.5 and 1.0 ratio of negative examples [**?**] did not provide stable evidence suggesting improvements for our task.

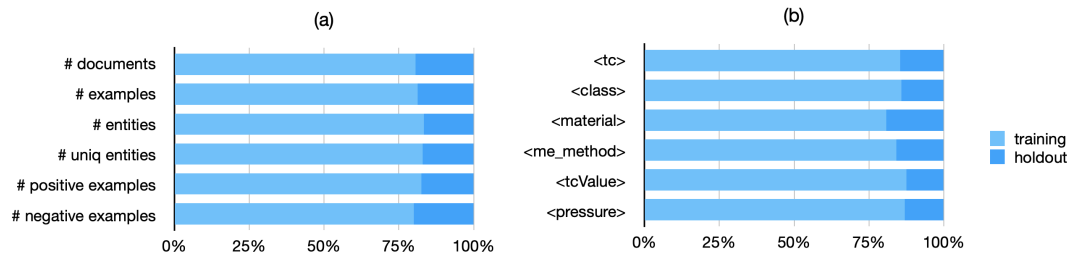

Figure 4. Holdout/Training set distribution (a) general metrics and (b) entities labels. *# entities* and *# unique entities* indicate the number of labelled entities with and without value duplicates, respectively. *positive examples* (+) indicates the number of sentences with at least one entity and *negative examples* (-) the number of sentences with no entities.

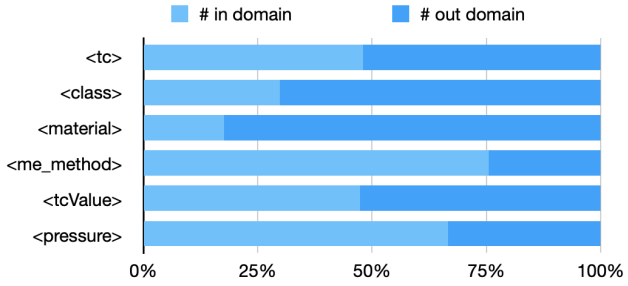

Figure 5. Holdout "out-of-domain" rates. Compares the amount of entities from the holdout set that are also in the training set (in-domain) with the entities that are not in the training set (out-of-domain)

Evaluation We report the evaluation results in Table **??**. The best results were obtained by Scibert with an F1 of 77.03%, and recall of around 80.69%. The features did not provide any improvements with RNN models: BidLSTM_CRF and BidLSTM_CRF_FEATURES resulted in the same F1 score. This result comes as a surprise because features such superscript/subscript were expected to be determinants for recognising materials sequences.

The `<pressure>` label obtained the lowest performances for all architectures, we believe 274 training examples are not sufficient considering that pressure expressions can be dependent from the context because they refer to outher type of pressures (e.g. annealing pressure). The labels with the highest score is `<material>`, with F1 of 80.77% and 78.06% for Scibert and BidLSTM_CRF, respectively. However, taking in account that such label has the highest "out-of-domain" ratio in the holdout set (above 75%, see Figure **??**) and the highest "label variability" (the ratio between unique entities and total entities, around 42%) this suggests that the model can cover well materials that has not been seen in the training. On the other hand the `<me_method>` label, which has lower "label variability" (around 11%) and low "out-of-domain" ratio, obtains an F1 score of 66.56% with Scibert and 65.92% with BidLSTM_CRF. Surprisingly, for label `<tc>`, the CRF is outperforming all the other architectures (F1 score of 83.96%), especially Scibert, which is down to 5 percentage points (78.35%). We can explain results by pointing out the extremely low variability (12.69%) of entities labelled as `<tc>` which is a more favourable terrain for the CRF.

Scibert also shows good generalisation capacity for unseen examples or examples appearing in a different context. In Figure **??**a only Scibert correctly extracts "above

100K", while CRF miss it completely and BidLSTM_CRF misses "above". In the training data "above 100K" is not present, however, we have "below 100K" and "1̃00K", and we have several other entities containing the token "above": "above 30K", "above 2K", etc. Scibert can understand that the token "above" is somehow relevant to the temperature. In Figure **??**b only Scibert can correctly extract "W-C nanowire" that is not present in the SuperMat training data ("out-of-domain" entity). Unfortunately, we cannot check whether "above 100K" or "W-C nanowire" are also present in the dataset used for pre-train SciBERT by their authors [**?**] because are not available.

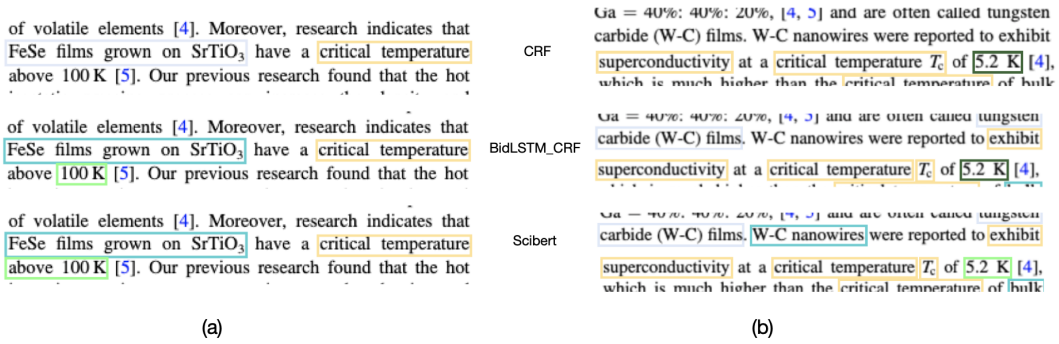

Figure 6. Examples [**?** **?**] of results from different architectures: CRF, BidLSTM_CRF and, Scibert

.

Label	CRF			BidLSTM_CRF			BidLSTM_CRF _FEATURES			Scibert			Supp
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	
<class>	79.74	66.79	72.69	79.01	72.62	**75.66**	77.84	72.40	74.97	72.95	75.28	74.09	1646
<material>	79	72.15	75.42	79.25	76.94	78.06	81.07	75.10	77.94	80.15	81.42	**80.77**	6943
<me_method>	60.25	68.73	64.21	56.41	79.49	65.92	55.86	80.45	65.90	56.26	81.52	**66.56**	1883
<pressure>	46.15	29.27	35.82	49.45	58.05	52.53	50.25	60.49	**54.36**	41.72	52.68	46.51	274
<tc>	84.36	83.57	**83.96**	78.61	82.54	80.48	79.19	82.07	80.60	74.46	82.66	78.35	3741
<tcValue>	69.8	66.24	67.97	70.36	75.16	72.67	68.95	76.56	72.52	70.90	79.74	**75.06**	1099
All (micro avg)	76.88	72.77	74.77	74.59	77.67	76.09	**75.17**	76.79	75.96	73.69	**80.69**	**77.03**	

Table 4. Evaluation scores for the superconductor ML model in the four architectures. For DL architecture the results are average over 5 runs. Support (Supp) indicate the number of labels in the training data.

Material ML model To train the *material ML model* we create a special dataset with an additional layer of labels (Table **??**) having as input the material information represented by entities annotated as `<material>` in the SuperMat documents.

Holdout set The annotations are performed on smaller chunks of text and the throughput is largely higher than the initial effort to develop SuperMat therefore we created an independent holdout set. We used material data extracted from a dataset of 500 documents (500-papers) from three publishers: *American Institute of Physics* (AIP), *American Physical Society* (APS) and *Institute of Physics* (IOP) [**?**]. The resulting holdout set has a average coverage above 25% (Figure **??**) and an "out-of-domain" ratio in average of 83.93% (Figure **??**).

Evaluation The results shown in Table **??** indicate that Scibert obtains the best results, with F1 at 84.15%. We confirm for this model that features can be ignored with

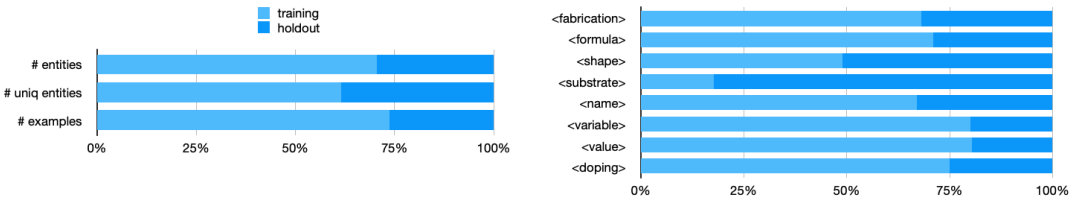

Figure 7. Holdout/Training set distribution for the *material ML model*. (a) general metrics and (b) entities labels.

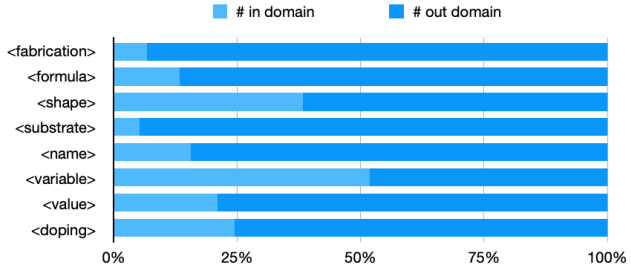

Figure 8. Holdout "out-of-domain" rates for the *material ML model*. Compares the amount of unique entities from the holdout set that are also in the training set (in-domain) with the entities that are not in the training set (out-of-domain)

the BidLSTM_CRF architecture, they improve the results by less than 1% (from 83.13 to 83.76%). The label `<fabrication>` does not perform well with any architecture, we believe, because it is too generic (Table **??**) and the content is too heterogeneous. Another label, `<substrate>` has 1/3 of the training examples of `<fabrication>` but obtains results 3 times higher with Scibert. This suggests to split `<fabrication>` into separate and more homogeneous labels.

Label	CRF			BidLSTM_CRF			BidLSTM_CRF _FEATURES			SciBERT			Supp
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	
<doping>	60.41	55.85	58.04	67.98	62.42	64.95	69.00	62.34	**65.43**	63.58	62.79	63.16	792
<fabrication>	40.00	4.55	8.16	23.61	5.91	9.24	37.33	9.09	14.48	22.51	13.18	**16.52**	94
<formula>	80.81	82.29	81.54	82.59	84.14	83.35	83.83	85.14	84.47	84.53	86.56	**85.53**	6301
<name>	72.2	63.75	67.71	76.29	78.76	77.43	74.51	80.38	77.33	77.18	81.86	**79.44**	1930
<shape>	90.89	92.51	91.69	90.93	95.79	**93.29**	90.33	95.74	92.96	89.67	97.20	93.28	809
<substrate>	37.04	6.76	11.43	54.31	32.43	40.44	60.08	33.38	42.82	56.32	41.22	**47.59**	32
<value>	80.21	83.15	81.65	84.81	89.33	86.99	85.16	90.15	**87.58**	83.14	85.92	84.50	1895
<variable>	96.85	95.98	96.41	95.19	97.77	96.46	96.32	97.90	**97.10**	96.22	96.52	96.37	1795
All (micro avg)	81.15	78.09	79.59	82.76	83.50	83.13	**83.20**	84.33	83.76	83.11	**85.23**	**84.15**	

Table 5. Evaluation scores of the material ML model with holdout set.

2.3. Entity Linking

The "Linking" aims to perform entity linking (EL) between materials and their corresponding properties.

We use a rule-based algorithm; other approaches such as the use of dependency parsing [**? ? ? ?**] were discarded. It was difficult to find a suitable dependency parser for scientific texts, and complementary methods based on complex rule sets were needed to compensate for the poor performance of the parser.

We link pair of entities focusing on three types of relationships:

- **material-tcValue** between material and its corresponding superconducting critical temperature value T_c.
- **tcValue-pressure** between superconducting critical temperature value and its related critical pressure.
- **me_method-tcValue** connects the superconducting critical temperature value to its corresponding measurement method.

Entities of type `<tcValue>` are pre-processed thorugh a classifier that establish if they are superconductors critical temperature T_c or not. This rule-based classifier combines the extracted entities of T_c expressions (label `<tc>`) with a set of predefined standard terms. When a T_c is not considered a "superconducting critical temperature" it is excluded from the list of possible linking candidates.

The linking rule-based approach works considering two scenarios: a) if entities to be linked have cardinality one in the sentence, they are linked automatically. When their cardinality is higher then if the word "*respectively*" appears in the sentence, we apply "order-linking", otherwise "distance-linking". For example, the sentence:

> P-or Ba-122 and Co-doped Ba-122 have lower T c s of about 30 K and 24 K, respectively, which makes helium free operation questionable.

containing the word "respectively", and applying "order-linking" we obtain: *P-or Ba122* is assigned to *30 K* and *Co-doped Ba-122* to *24 K*.

The other approach "distance-linking" works by defining the distance measurement d as a value calculated in numbers of characters between the centroid of each entities. Entities surrounded by parenthesis are expanded it to the whole parenthesis and its centroid is updated. As an example, in the sentence

> We tested two materials MgB2 (Tc = 39 K) and FeSe (Tc = 16 K).

`39 K` is closer to `FeSe` (d=10) than to `MgB2` (d=11). Both temperatures entities are expanded to their containing parenthesis e.g. `39 K` to `(Tc = 39 K)` in this case the centre of the entity `39 K` is shifted toward the left, from the initial value of 38 to 35 and the distance from `MgB2` is reduced from d=11 to d=8. As a result, the `MgB2` entities is correctly linked to `39 K`.

The distance calculation is also adjusted with the addition of "penalties" by doubling the calculated distance when certain keywords such as ",", ".", ";", "and", "but', "while', "whereas', "which", "although" representing logical separation of predicates appear between the two entities [**?**]. In the above example, the distance between `39 K` and `FeSe` will be doubled (d=20).

The rule-based linking is evaluated using the linked entities from SuperMat [**?**] (Table **??**). Each task aims evaluating the linking between two entities types, assuming a 100% accuracy in the previous step. The result of the `material-tcValue` indicates F1 score around 80% with a precision of 88.40%.

Link type	**Precision**	**Recall**	**F1-score**	Support
material-tcValue	88.40	74.52	80.87	726
tcValue-pressure	85.71	71.52	77.98	118
me_method-tcValue	62.28	65.74	63.96	151

Table 6. Evaluation scores for the Linking.

2.4. End to end evaluation

The end-to-end evaluation (E2EE) measures the capacity of the system on unseen documents. We limit the scope of the E2EE to the triplet 'material-Tc-pressure' which, at the moment, are the backbone upon which the database is built. We perform the E2EE on the 500-papers dataset where we manually examine the resulting database as follows: a) we mark invalid records and b) we identify the cause of failure from a predefined set of five *error types* (Figure **??**):

- **From table**: the extracted text is wrongly extracted from a table. Although table content is ignored, the error rate from the Grobid library is still relevant due to the lack of training data.
- **Extraction**: the failure was caused by entities not recognised, wrongly or partially recognised.
- **Quantities extraction**: the entities of type quantities (pressure, temperature) are not correctly extracted. We measured it separately to identify a failure on a separate ML model.
- **Tc classification**: the temperature is wrongly classified as superconducting Tc
- **Linking**: given the precedent steps were performed correctly, the resulting entities were not linked correctly.

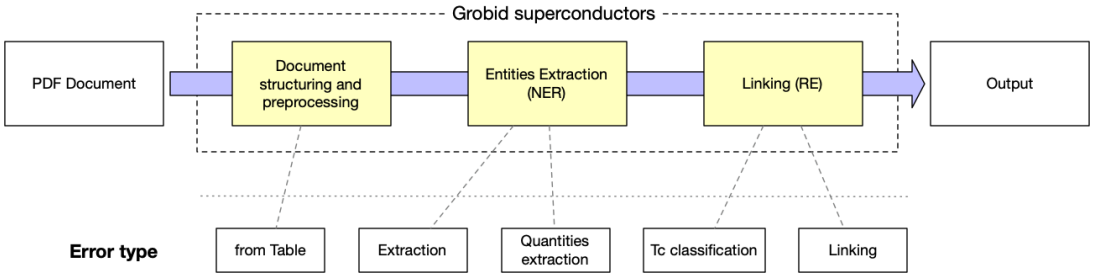

Figure 9. *Error types* in the context of the data flow.

The E2EE scores are illustrated in (Table **??**). The recall is omitted because less relevant and difficult to calculate manually. The system scores 72.60% precision considering all the subsections. Comparing precision by subsection, we notice a clear difference between the error rates of Figure caption (59.28%) and unknown subsections (57.14%) with the rest of the other subsections ($> 70\%$). Unknown*subsections* indicate that the extracted text was not well identified by Grobid but it was aggregated nevertheless. The scores increase to 73% when excluding unknown subsections, 75.24% when excluding figure captions, and 79.14% when excluding both. Excluding these two subsections will not impact the amount of text, because both count for less than 20% of the total number of subsections.

The error types are summarised in Figure **??**. The most common failures originate from Tc classification (40%), Linking (32%), and Extraction (20%).

The most common Tc classification failures are as incorrect recognition of a) relative values of T_c(eg. 1 K higher than material X), b) values indicating the transition temperature width (ΔT_c), c) temperature values that are not T_c, for example, material synthesis temperatures T, other critical transition temperatures that are not superconducting (e.g T_{Curie}), and d) values of temperature at which there is no superconductivity (e.g at 70 K there is no superconductivity). The errors of type "Linking" occurs mainly when the authors are comparing relative values of T_c using materials

Subsection	Precision	Support
Title	100	2
Abstract	80.32	61
Paragraph	75.2	623
Figure captions	59.28	140
Unknown	57.14	21
Micro avg.	72.60	847
Micro avg. (excl. figures)	75.24	707
Micro avg. (excl. unknown sections)	73.00	603
Micro avg. (excl. figures and unknown sections)	79.14	657

Table 7. Evaluation end to end: summary of the scores.

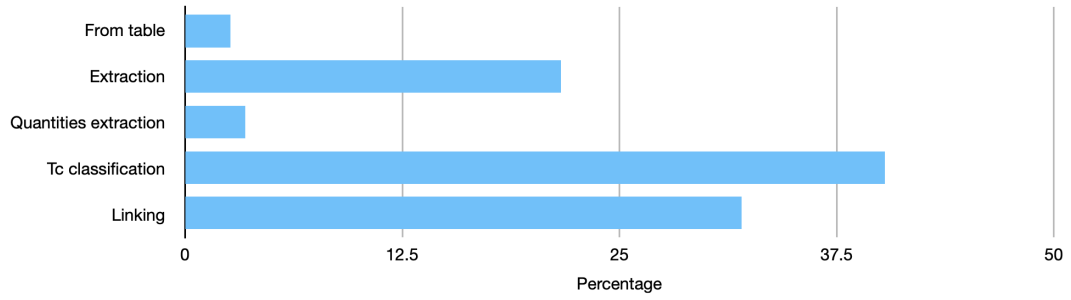

Figure 10. Error type distribution in the E2EE on the *500-papers* dataset.

for comparison (e.g. *The Tc = 38 K is similar to the one of* MgB_2). Finally, "Extraction" issues originate mainly from: a) implicit mention of the main material when experimented with different substrates, and b) mismatches between `<material>` and `<class>` which, by definition, overlap.

3. Supercon^2

We created SuperCon^2 by processing 37770 research papers belonging to the category *cond-mat.supr-cond* in ArXiv. Currently SuperCon^2 contains 40324 records including 2052 triplets *material-Tc-applied pressure*, and 3602 records with explicit measurement method *material-Tc-measurement method*. The schema of SuperCon^2 is summarised with examples in Table **??**.

The data is ingested through the asynchronous Map-Reduce approach [**?**]. The "Extraction task" processes the PDF documents with *grobid-superconductors* and stores their processed representation together with the original PDF document. Furthermore, the "Aggregation task" reduces the document information in a synthesised tabular format. We store the processed document representation in JSON format, and they are keep separately to being used for visualisation together with the original PDF. The pipeline uses a persistence layer for storage and reporting (logger) accessing *grobid-superconductors* via REST API.

We built a visualisation interface to exploit the extracted information. Users can search in the synthesised tabular data, access the PDF document enriched with the extracted information (Figure **??**), and export locally in CSV, TSV and Microsoft

Excel formats.

Field name	
Material information	
Raw material	The material or sample as
Name	Canonic
Formula	Material expressed as c includes also formulas with s
Doping	Doping ratio that might be adj
Shape	The shape of the n
Variables	Variables that can be subs
Class	
Fabrication	All the infor belong to an
Substrate	Substrate material describe
Properties	
Critical Temperature	Superconductin
Applied Pressure	Pressure ap the superconductin
Measurement Method	Method fo superconductin
Document bibliographic information	
Section	The main bod
Subsection	The secondary segmenta
Authors, Title, DOI, Publisher, Journal, Year	Bibliographic informa
Internal information	
Hash, Timestamp	Hash calculated on the binary content of the original PDF document and the timestamp when the doc

Table 8. Summary and description of the SuperCon^2 schema. *Internal information* are technical information not accessible to the users.

4. Conclusion

In this work, we present our solution for automatically building a database of materials and properties from scientific literature. Our contribution is composed of: a) *grobid-superconductors*, a specialised open source system that processes PDF documents combining ML and rule-based methods to extract and link relevant information in superconductors research. b) A pipeline allowing large-scale document processing, and c) a visualisation interface for rapid data exploration which includes PDF documents information enrichment.

We made SuperCon[2], a database with 40324 records of superconductors materials and properties including the applied pressure and the T_c measurement method. SuperCon_2 is available in text format at `https://github.com/lfoppiano/supercon`.

In future, we plan to improve our tools by a) extracting more properties, such as crystal structure type, space groups type, and lattice structure, b) train supervised models for the Linking step, and c) extending the interface to support data correction toward efficient curation. We confirmed the good generalisation ability of the Scibert architecture for NER task in materials science domain. There are hopes to obtain better results using materials science pre-trained BERT, such as matscibert [**?**], however, the gain might be just minimal for relatively larger models [**?**].

Acknowledgement

Our warmest thanks to Patrice Lopez, the author of Grobid [**?**], DeLFT [**?**] and many other interesting open-source projects.

Competing interests

The authors declare no competing interests.

Appendix A. Dataset additional information

	training	**holdout**	**holdout/training**
# documents	132	32	24.24%
# examples	16902	3905	23.10%
# entities	15586	3112	19.97%
# unique entities	6699	1372	20.48%
# positive examples	8380	1776	21.19%
# negative examples	8522	2129	24.98%

Table A1. Holdout/Training set distribution between training and holdout sets for the *superconductors ML model*. *# positive examples* indicate the number of sentences with at least one entity, and *# negative examples* the number of sentences with no entities.

Appendix B. Machine learning support material

label	**training**	**holdout**	**holdout/training**
<tc>	3741	639	17.08%
<class>	1646	271	16.46%
<material>	6943	1649	23.75%
<me_method>	1883	355	18.85%
<tcValue>	1099	157	14.29%
<pressure>	274	41	14.96%

Table A2. Holdout/Training set distribution between training and holdout sets on different labels for the *superconductors ML model*.

	training	**holdout**	**holdout/training**
# examples	13648	5728	41.97%
# entities	4512	2817	62.43%
# unique entities	9268	3292	35.52%

Table A3. Holdout/Training set distribution training and holdout sets for the *material ML model*.

label	**training**	**holdout**	**holdout/training**
<fabrication>	94	44	46.81%
<formula>	6301	2569	40.77%
<shape>	809	841	103.96%
<substrate>	32	148	462.50%
<name>	1930	949	49.17%
<variable>	1795	449	25.01%
<value>	1895	463	24.43%
<doping>	792	265	33.46%

Table A4. Holdout/Training set distribution training and holdout sets on different labels for the *material ML model*.

#	**Feature**	**Model**	**Architecture**
1	current token	all	all
2	current token lower cased	all	all
3-6	(four features) current token, prefix characters 1 to 4	all	CRF
7-10	(four features) current token, suffix characters 1 to 4	all	CRF
11	information about capitalisation: first character (INITCAP), all characters (ALLCAPS), none (NOCAPS)	all	all
12	digits content: all (ALLDIGIT), some digits (CONTAINDIGIT), no digits (NODIGIT)	all	all
13	(boolean) the token is composed by a single character	all	all
14	punctuaction information and normalisation to placeholders: no punctuation (NOPUNCT), open or end brackets (OPENBRACKET, ENDBRACKET), various punctuation (DOT, COMMA, HYPHEN, QUOTE), open or close quotes (OPENQUOTE, ENDQUOTE), anything else (PUNCT)	all	all
15	Shadow the numbers	all	CRF
16	Shadow any characters: “x” for lowercase, “X” for uppercase, “d” for digits	all	CRF
17	As the previous but compressed	all	CRF
18	Font name	superconductors	all
19	Font size	superconductors	all
20	Font style: standard (BASELINE), superscript (SUPERSCRIPT) or subscript (SUBSCRIPT)	superconductors	all
21	(boolean) if the token style is bold	superconductors	all
22	(boolean) if the token style is italic	superconductors	all
23	(boolean) the token is identified as a chemical compound by ChemDataExtractor[**?**]	superconductors	all

Table B1. Summary of the features used in the *superconductors* and *material* ML models. *All* under Architecture indicate only BidLSTM_CRF_FEATURES and CRF.