Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-021-01807-x

"FabNER": information extraction from manufacturing process science domain literature using named entity recognition

Aman Kumar[1] · **Binil Starly**[1]

Received: 19 April 2021 / Accepted: 17 June 2021

Abstract
The number of published manufacturing science digital articles available from scientific journals and the broader web have exponentially increased every year since the 1990s. To assimilate all of this knowledge by a novice engineer or an experienced researcher, requires significant synthesis of the existing knowledge space contained within published material, to find answers to basic and complex queries. Algorithmic approaches through machine learning and specifically Natural Language Processing (NLP) on a domain specific area such as manufacturing, is lacking. One of the significant challenges to analyzing manufacturing vocabulary is the lack of a named entity recognition model that enables algorithms to classify the manufacturing corpus of words under various manufacturing semantic categories. This work presents a supervised machine learning approach to categorize unstructured text from 500K+ manufacturing science related scientific abstracts and labelling them under various manufacturing topic categories. A neural network model using a bidirectional long-short term memory, plus a conditional random field (BiLSTM + CRF) is trained to extract information from manufacturing science abstracts. Our classifier achieves an overall accuracy (f1-score) of 88%, which is quite near to the state-of-the-art performance. Two use case examples are presented that demonstrate the value of the developed NER model as a Technical Language Processing (TLP) workflow on manufacturing science documents. The long term goal is to extract valuable knowledge regarding the connections and relationships between key manufacturing concepts/entities available within millions of manufacturing documents into a structured labeled-property graph data structure that allow for programmatic query and retrieval.

Keywords NER · Technical language processing · TLP · Word2Vec · Topic modeling

Introduction

Over the past decade, the plethora of manufacturing processes developed across a range of application domains are considerable. Yet, much of the knowledge linking product design with manufacturing resides within the minds of experienced professionals or documented in the form of books, magazines, scientific articles and multi-media. For untrained professionals, searching for specific information about design and manufacturing will require online search through multiple digital media formats. Since most search engines do not understand the context surrounding manufacturing related query terms, results produced by the search engines rely on pure text-based indexing (Gusenbauer, 2019). Users must manually parse through tens of hundreds of search results links to find the information they seek. With the exponential rise in number of manufacturing related articles and associated digital resources available on the web, searching for and extracting valuable information is challenging. Finding specific concepts or related concepts from within scientific articles and other digital media require manual processing by domain experts to synthesize. Even then, such manual processing requires tedious parsing across multiple forms of available media to make informed decisions. As opposed to parsing natural language text, this subdiscipline within NLP, would be considered as Technical Language Processing (TLP) (Brundage et al., 2021), i.e., the process of extracting useful information and knowledge from technical documents.

Such automated information and relationship extraction (Zheng et al., 2017) is critically important to a wide variety of users. This includes manufacturing focused researchers,

✉ Binil Starly
bstarly@ncsu.edu

[1] Edward P. Fitts Department of Industrial and Systems Engineering, North Carolina State University, 915 Partners Way, Raleigh, NC 27606, USA

Published online: 24 June 2021

young engineers and non-technical professionals. New modes of interactions with technical content can broaden the inclusion of those that are involved in the product design and manufacturing space, specifically non-technical entrepreneurs and physical product focused startups. The system can also benefit businesses seeking to diversify product supply chains (Shahzad & Hadj-Hamou,2013; Sharma et al., 2020; Zuzarte & Proença, 2019), employ chatbot (Chatbot, 2007), hire required workforce (Mittal et al., 2020), plan hazard management (Brewer et al., 1999; Kung et al., 2020) automatically analyze patent literature (Abiodun et al., 2018) for new recommendations and even targeting custom advertisements relevant to the content of a digital magazine (Shah et al., 2020). None of the existing natural language processing techniques is suitable to extract relevant information from the manufacturing context domain. Existing techniques are either trained on general text for identifying information such as Person, place, organizations, location etc. or on domain specific texts such as Legal, Bioscience, and Finance etc. Besides beyond just named entity recognition applications, automatic analysis of manufacturing literature can also power Question-Answering (QA) systems in manufacturing education (Cui et al., 1903; Lende & Raghuwanshi, 2016), power chat bots specific to manufacturing business data (Kassner et al., 2017) and troubleshoot technological problems (Alfeo et al., 2021).

This research work is mainly focused towards extracting information from manufacturing process science text abstracts using Named Entity Recognition (NER), as a text mining technique on a large corpus of scientific manufacturing process science abstracts. The developed model learns to identify manufacturing process science keywords within a sentence in a scientific abstract and maps them to specific manufacturing topic categories. To the best of our knowledge, no previous NER model has been built to classify manufacturing process science text. Our work describes the annotation of the data, techniques to tag the data and the necessary training of word embeddings and associated neural network architectures. Two use cases of the NER model is presented that demonstrates the value of NER towards technical language processing.

Related work

Named entity recognition (NER) is one of the important text-mining subtasks towards information extraction. Existing standard methods for NER focus on extraction of general entities such as persons, organizations, locations etc. from within a sentence (Nadeau & Sekine, 2007). One of the best applications of NER is the article recommendation systems by leveraging the entities appearing in the current article, and then finding similar articles that have similar number of entities (Li et al., 2020a). NER has quite varied applications in every domain, and the potential use cases of it are being studied across diverse domains such as material science (Weston et al., 2019), biomedical sciences (Leaman & Gonzalez, 2008), chemical science (Rocktäschel et al., 2012), cybersecurity (Gasmi et al., 2018), maintenance (Navinchandran et al., 2021) and forensic science (Studiawan et al., 2018). Using NER, one can extract entities in a sentence to understand the context of the sentence topic without the prior knowledge. NER is also one of the critical steps towards development of a Knowledge Graph, and among the reasons on why it has become quite popular (Costa et al., 2016; Kejriwal, 2019).

The standard NER model would not be able to identify the terms and concepts for a specific domain such as manufacturing. Domain-specific NER has been quite prevalent, and it has been applied to many diverse domains but these models have not been tested against the corpora of words/phrases utilized within the manufacturing process science text. Previously, work by Weston et. al. focused on Named entity recognition for material science which claimed to extract the summary level information from research papers, and extraction of material and entities in seven categories. They have used a neural network model consisting of a recurrent neural network (RNN) architecture i.e. BiLSTM along with CRF model for word and character-level feature recognition. A chemical named entity recognition by Safaa et. al. mentions different methods of NER approach such as dictionary-based, rule-based and hybrid forms in identifying chemical substances (Eltyeb & Salim, 2014). In other similar work in this domain by Jiaguan et al., the authors mention the 10 different types of entity extraction of chemical reactions from patents (Nguyen et al., 2020). A work related to medical knowledge graph (Li et al., 2020b) talks about the application of NER in Electronic medical records (EMR) processing algorithm, and identification of medical entities such as medicines, diseases and symptoms, from within the records. NER finds its application in mechanical engine fault knowledge extraction (Chen et al., 2020) as well as replacing the traditional research methods of using structured data to process recognition of engine fault related knowledge from unstructured text. The work by Hehua (Yan et al., 2020) is a close work related to manufacturing NER, but it lacks concrete mention of the specific entities categorization. For NER extraction, they have opted a machine learning based method along with CRF model. Lastly, Orcun proposes that NER can improve the semantic question answering for smart factory domain (Oruç & Aßmann, 2020) by assisting in extracting the entity-relationship pair.

In our current analysis, there is no comprehensive study on building an NER model to automatically classify manufacturing process science text, particularly at a large scale. Such an NER is critical to interlinking manufacturing concepts and to

integrate new knowledge concepts to existing knowledge. Such NER models are also critical to building technical language processing of manufacturing text related to asset management, product design and manufacturing, quality inspection, critical safety reports and developing factory-based chat bots specifically to aid the manufacturing workforce. For those trying to understand a new manufacturing process domain, such as metal additive manufacturing, it would be extremely useful for researchers and industry practitioners to ask questions of a meta-analysis nature from all published scientific literature, such as—"which academic universities work on copper as a metal additive manufacturing process" or "what materials have been studied for use in metal additive manufacturing, and filter out those that have been studied in the context of medical implants". Answers to these queries require humans to tediously parse through hundreds and thousands of articles, before an answer can be produced, even after text based indexing performed by scientific article publishing companies. However, by building algorithms to analyze the entire corpus of available manufacturing process science literature, it is possible to represent published articles to unique database identifiers, which then can be programmatically queried to retrieve answers to such user queries.

In this work, we build an NER model for large-scale information extraction from the manufacturing process science literature. This NER model is capable of parsing through more than 500K + relevant scientific abstracts and classify recognizable entities within the abstract as belonging to 12 categories—material, process, machine/equipment, application, engineering features, mechanical properties, process characterization, enabling technology, concept/principles, manufacturing standards and biomedical. We have focused building and testing the model on text available within scientific abstracts primarily due to two reasons—ease of accessibility as those mode available by Web of Science and second due to the concise representation of words to convey the scientific study represented in the paper. The NER model is a neural network model trained using 1200 + hand-annotated abstracts. We have obtained an overall f1 score accuracy of 88%. We demonstrate how this trained NER model can be used as a topic extraction exercise from a given paragraph of manufacturing process text and how similar and related words can be grouped together based on a query term of interest. In addition, the dataset containing hand-annotated datasets of various manufacturing terms classified under the 12 categories is made available to the community.

Methodology

The overall architecture of the work is shown in Fig. 1. Each step of the process is described in detail, along with the analysis of the extracted results in the two specific use cases of the model. Machine learning models used in this work were built using open source libraries through scikit-learn, gensim, Tensorflow and keras libraries.

Data collection and pre-processing

This work focuses on the large-scale text mining of the manufacturing process science abstracts. A total of 500K+ abstracts were obtained from Web of Science through known journals available in manufacturing process science research. These include—Journal of Manufacturing Processes, J. Manufacturing Systems, Additive Manufacturing, Rapid Prototyping, Advanced Materials, Int. Journal of Adv. Manufacturing Technology, ASME J. Manufacturing Science etc. In addition, manufacturing keywords were also used to gather abstracts within the manufacturing process domain. Abstracts were selected between the year 2000 and 2020. Abstract retrieval using search terms such as "3D printing", "Additive manufacturing", "Titanium alloy", "Machining", "Welding" etc. were used along with domain specific journals to ensure that the corresponding article would help generate enough abstracts within a domain. More abstracts could have been collected but were limited due to access and download restrictions. Since our initial collection yielded sufficient results to help train the NER model, collecting more abstracts may not add significant value to the training and inference process. The reason behind addition and accumulation of abstracts in the corpus was to take the gist of research papers in a brief text that will provide quality and enough vocabulary variability. Short and important domain specific text helps in improvement of overall quality of the corpus, and removal of clutters. It was demonstrated by Tshitoyan et al. (2019) that models trained on unrelated text performs poorly on word embedding related activities such as fetching similarities/analogies of words.

Manufacturing text categorization

The extraction of specific types of entities can assist in knowing the overall context/summary of any given text. For every word, there were categories/entity labels defined namely Material (MATE), Manufacturing Process (MANP), Application (APPL), Features (ENGF), Mechanical Properties (MECHP), Characterization (PROC), Parameters (PROP), Machine/Equipment (MACEQ), Enabling Technology (ENAT), Concept/Principles (CONPRI), BioMedical (BIOP) and Manufacturing Standards (MANS). The category names and descriptions are given in Table 1.

Tagging process (tokenization)

To generate the manufacturing keywords to be annotated by humans and labelled under the 12 categories, multiple

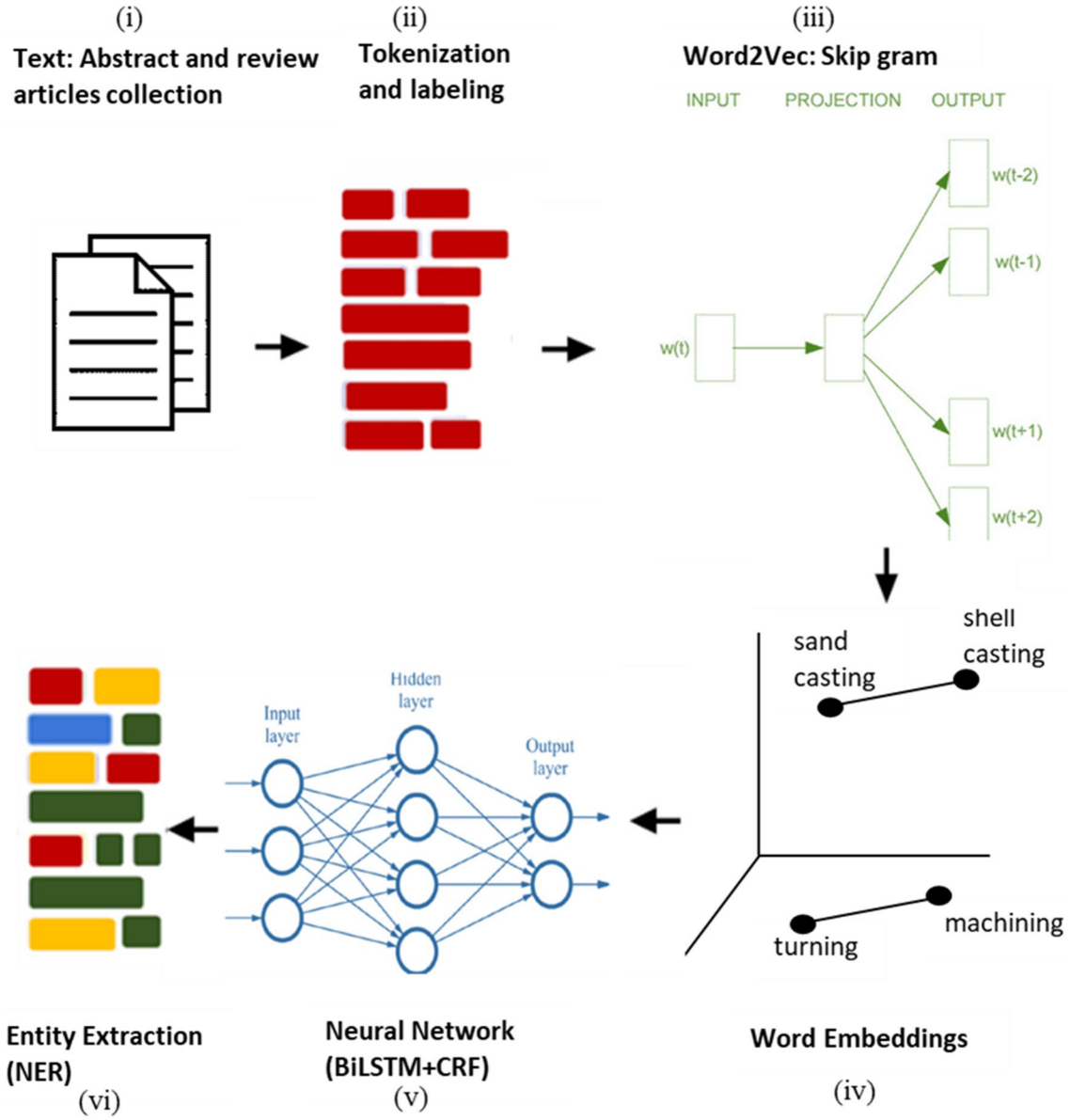

Fig. 1 Methodology for named entity recognition involves (i) collection of abstracts and preprocessing them to develop a corpus of cleaned data, (ii) tokenization of every word in a small subset of the corpus and manually labeling of entities in respective category for developing training data, (iii) use of word2vec skipgram technique for (iv) creating context based embeddings on whole corpus, (v) training the neural network model using the embedding and (vi) finally extraction of named entities

sources were utilized to cover the breadth of the manufacturing process science text. The indices behind two textbooks, keywords from the manufacturing domain research papers and vocabulary words from 75 manufacturing process science abstracts were utilized for the human labeling process. Four graduate students with an appropriate background in manufacturing process science were selected for the human labelling process.

The abstract papers and review paper terms were tokenized one line per sentence, and textbook index were taken in a separate file having one entity per line. The annotation process began first by utilizing the keyword indices available in the manufacturing textbooks to yield a dictionary of labeled entities. In the second stage, the annotation was carried out on tokenized sentences from the 75 abstracts and the entire words list available in selected review articles. While tokenizing each line per sentence, the entities that appeared in textbook index dictionary were automatically labeled. The remaining words were divided up among the human labelers. Even though some terms were automatically tagged due to its appearance during index term tagging, it was important to go through every token to check

Table 1 Named entity categories and descriptions

S no	Categories	Example
1	Material	Such as stainless steel, titanium alloy
2	Manufacturing process	Such as additive manufacturing, forming, shaping, joining
3	Machine/equipment	Such as lathe, CNC machine, grinding machine, confocal sensor
4	Application	Such as industry—aerospace, automotive, medical, pharma etc
5	Engineering features	Such as cut, extrude, slots, sweep, loft, radius, fillet
6	Mechanical properties	Such as corrosion resistance, yield strength, hardness
7	Process characterization	Such as CT scan, x ray diffraction or any other measurement technique
8	Process parameters	Such as cutting speed, spindle speed, laser power, MRR
9	Enabling technology	Such as blockchain, MT connect, CAD, OPC/UA
10	Concept/principles	Such as abrasion theory of friction, Smart Manufacturing
11	Manufacturing standards	Such as STEP, STL, ISO classification, ANSI, DIN, ASTM etc
12	Biomedical	Such as trabecular bones, scaffolds, starch, tendons etc

if the dictionary word precedes or supersedes with another entity, as it will change the tagging of the respective entity. For ex., if textbook index had the term 'sand' which was automatically tagged for one token of material category, but if another term in the tokenized sentence from research papers that follows is 'casting' which is a manufacturing process, the words 'sand casting' would make it classified under a completely different category—which is a "manufacturing process" category. Once each labeler completed their individual assigned annotation tasks, the categorization was verified by another human. We recorded the number of instances in which a discrepancy was obtained between the initial annotator and the verifier. One noted challenge was that vocabulary words that consistently re-appeared in the text had to fall under the same category or else this would introduce errors in the neural network training process. Any such discrepancies were removed before the training process had begun. The token-based approach was tedious and time-consuming.

Tagging process (vocabulary extraction)

An alternate approach is based on vocabulary-based splitting of sentences to account for words that appear in sentences within the various available datasets assigned for training purposes. The challenge here is to extract the vocabulary words automatically in the first place for the annotation process. We chose parsing of research articles and utilized the method described by Michael et. al. (2019) to identify terms with corresponding part-of-speech (POS) tags, chunked noun and verb phrases using the NLP tool SpaCy (Honnibal & Johnson, 2015). We then performed this method by developing a corpus of 1200+ abstracts and 20 review articles, and a separate index from 3 textbooks. This methodology was quite effective in terms of filtering clutters and narrowing down the entities phrase from review papers upon removing the duplicate terms that were extracted from textbooks and research paper keywords. Later, the entity phrases were edited and verified with human supervision to avoid overseeing of any important entity. Following this, we were able to gather 11,000+ uniquely related entities, which were then subsequently annotated by humans to fall under the 12 categories. These entities are the manufacturing dictionary words following the procedure mentioned above. An error drop of about 8% was observed following this new method when compared to the token-based annotation. Table 2 mentions the number of entities that were incorrectly annotated and how that dropped when vocabulary-based method was adopted.

BIOES tagging scheme

The task of named entity recognition is to assign every word in a sentence to a named entity label. All the terms were compiled in a dictionary, and for each token in the corpus, annotation was performed in all categories along with the output tag in 'BIOES' format: B=Beginning, I-Intermediate, O=Outside, E=End, S=Single. This format assists in accounting for manufacturing related phrases such as 'Metal Additive manufacturing' or 'Casting'. Here, Metal Additive manufacturing will be labeled as 'Metal: B-MANP; Additive: I-MANP; Manufacturing: E-MANP' where B denotes

Table 2 Entities annotation accuracy

Entities annotation type	Total entities	Annotation discrepancy	% Annotation discrepancy
Initial textbook index words	280	27	9.64
Initial tokenized annotation	3534	345	9.76
Final vocabulary annotation and verification	11,432	201	1.76

the beginning of the phrase, 'E' is the end of the phrase and 'I' is everything that is within 'B' and 'E', and the word 'Casting' will be labeled as 'S-MANP'. The BIOES tagging scheme has been documented to outperform (Ratinov & Roth, 2009) others, such as BIO which stands for 'B=Beginning I=Intermediate O=Others'. BIOES notation gives more detailed description of every token position in the entities. In the matched entry, BIOES also indicates the token's location. In other words, B or E will not appear in a partial match with only a suffix. These human labelled annotations assist in training algorithms to understand the context and the sequence of words used. Existing tagging methods in natural language processing (NLP) such as regexner from the Stanford coreNLP (Manning et al., 2014) was not used since the technique did not support tagging across schemes that support the BIOES or BIO tagging format. In domain specific language such as in manufacturing, it is common to find bi-gram or even tri-gram words which convey a meaning that is quite different if these n-gram words are split to individual words. Examples, such as 'Metal Additive Manufacturing' or 'Titanium alloy' refers to a specific manufacturing process and material respectively. However, if these n-grams were to be split into individual words, they convey a completely different meaning, which can lead to existing tagging methods to improperly tag them, which eventually leads to ambiguous results.

With the expanded vocabulary covering most of the terms, the customized vocabulary extraction algorithm was used to automatically label the corpus in BIOES format. Once done, experiments were performed to check the ratio of 'O' label with respect to the length of the sentence. Any sentence with a ratio of 0.9 or less was kept intact and filtering out the ones above this value. This ratio is the representation that out of 10 words in any given sentence, at least one is labeled as a non 'O' category. This will ensure that there are quality of sentences having tags provided as training to the neural network. Sentences that do not have a non- 'O' category is not helpful in model training. The ratio was varied from 0.85, 0.8, 0.75, 0.7, 0.65 and 0.6 but even though decreasing the ratio increases the quality of sentences with NER, it starts decreasing the variation of entities that are tagged due to some entities being lost due to it appearing in sentences with a lesser ratio. Figure 2 shows how the number of entities vary with respect to ratio of 'O' labels and number of words in sentence. Using the method mentioned above, in total we compiled a manufacturing text document of 350,000 + words for training the neural network for NER. The entire annotated dataset is made available as Supplemental Material (Kumar & Starly).

While automatically tagging of entities with respective tags, there were inconsistent tagging in some token words. This limitation of vocabulary-based tagging method was in the form of variations in usage of a single term such as the manufacturing process—'wire and arc additive manufacturing' was mentioned in the corpus in different ways, for ex. Wire + arc + additive manufacturing, or 'wire and arc additive manufacturing'. To counter this, we implemented Levenshtein distance between every two strings/phrase while automatically tagging entities in our algorithm (Haldar & Mukhopadhyay, 2011). The distance of Levenshtein between two strings is the number of deletions, insertions, or substitutions needed to convert the source string into the target string. Various threshold Levenshtein distances were experimented with, starting with a maximum of 2 and 3. This metric enabled us to capture terms such as 'additive manufacture' and 'additive manufacturing' or variations of

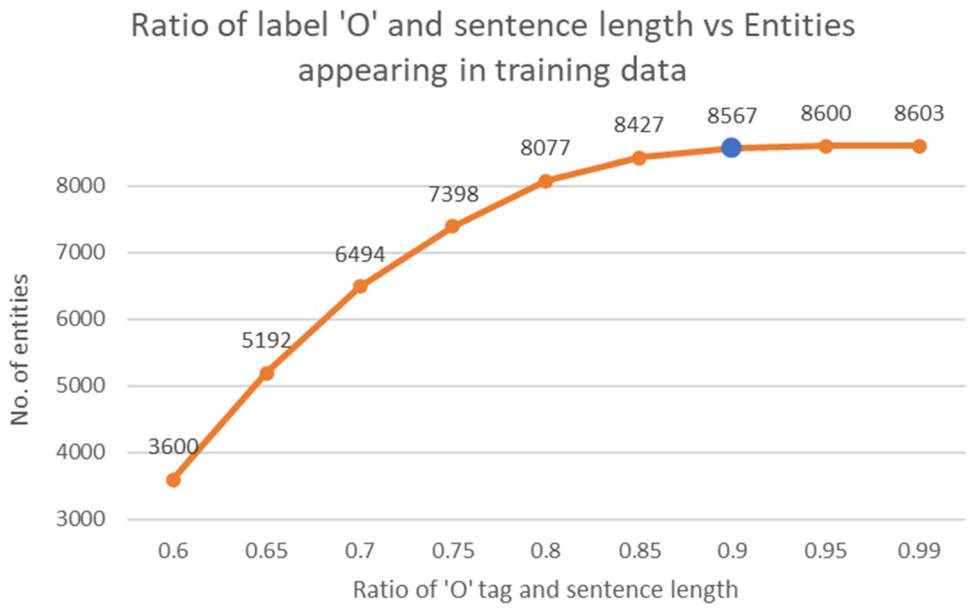

Fig. 2 Ratio of tag 'O' and sentence length versus number of entities appearing in training data

'wire and arc additive manufacturing'. However, terms such as 'range' which is a tagged entity 'parameter' got matched with a phrase 'a range' due to 2 Levenshtein distance. Similarly, every entity that was accompanied by a determiner or verb or adverb was being captured. To rectify this, we filtered out any words such as 'a, an, the' while calculating Levenshtein distance between tokens using regex (regular expressions library).

Neural network architecture

For this work, the architecture devised by Lampe et al. (2016), with a focus on bidirectional long-short term memory (LSTM) network and conditional random fields (CRF) was employed. This type of model relies on two information sources: 1) character-based representation of words learned from the supervised corpus, and 2) unsupervised representation of words learned from the un-annotated corpora. The corpus of manufacturing text was split in a ratio of 80:10:10 for training, validation and test set respectively. The objective of training the model using the architecture (BiLSTM + CRF) is to be able to recognize the words in the context of the manufacturing such that the model will provide us with the desired categories of the named entities. For instance, words such as maraging steel and PLA are marked as materials, while machining and nanofinishing (Kumar et al., 2019) lies in the category of manufacturing process. There are three key aspects of information which are important to train a model for the identification/recognition of named entities for specific category. These are (a) representation of word (b) context of sentence around the word (c) representation of character.

Word embeddings for representation of words

It is often said that the words are known by the company they keep, which is based on a distributional hypothesis (Goldberg & Levy, 2014) that the words appearing in a similar context have identical meanings. In natural language processing, the words in a text corpus are represented as numerical vectors in a high dimensional space such that their syntactic and semantic relationships remain intact. This representation technique is termed as vector space modeling or word embeddings. Word2vec, Glove (Pennington et al., 2014), ELMo (Peters et al., 2018), BERT (Devlin et al., 2018) are one of the most famous word embeddings methods.

To train the word embeddings, a corpus for manufacturing related text was created and processed as mentioned in the previous section. Once the text was collected and preprocessed, we used the skip-gram variation of Word2vec to our text corpus while training the word embedding on 300-dimensional vector space. The word embeddings upon training on additive manufacturing text corpus ensures that all words that are in vicinity to the target word in the corpus will produce vectors in a way that the cosine distance (Levy & Goldberg, 2014) between the words such as "stainless steel" and "sl316" would be very near i.e. closer to 1 (where 1 corresponds to same vector), while the word "polymer" will be comparatively farther away. The target terms 'stainless steel' and 'maraging steel' are represented as vectors of ones at their respective vocabulary indices and zeros everywhere else. These one-hot encoded vectors are used as inputs for a single linear hidden layer neural network (for example, 300 neurons) that is trained to predict all words from the given target word within a certain distance (context words). Table 3 shows all the parameters along with their explanations that were used to train the word embedding model via the python based 'gensim' library (Řehůřek & Sojka, 2011).

Various vector operations such as addition or subtraction of vectors can be performed to analyze how different words are related to one another. Domain specific analogies can be obtained just similar to what is depicted in the methodology of validating the word vectors (Mikolov et al., 2013). In the Word2vec model, such analogies are represented and solved by seeking the closest word to the outcome of addition and subtraction operations in the embeddings. For example, addition of 'drilling' and 'micromachining' gives 'microdrilling' as the result which is correct, as microdrilling is the drilling performed at a micro level. Another example, addition of 'welding' and 'stainless steel' gives 'tungsten inert gas welding' that is correct as TIG welding is mainly used for welding of stainless steel.

To validate the word embeddings, experiments were performed by separating some common acronyms used within the manufacturing process and materials processing literature. We took a random acronym set of about 70 entities, which mainly contained a set of manufacturing processes, materials, property, and other entities category. Out of 70, 52 were found by the word similarity method within the top 5 similar results while 5 were not found, as shown in Table 4. Out of the terms not found, the acronym 'mig', having abbreviation 'metal_inert_gas', was not found near to each other in any sentence in the corpus while doing keyword-based search. Similarly, for the acronym of 'saw' as 'submerged_arc_welding', results related to 'sawing' and 'saws' were observed. The reason behind any acronym not showing as a top 5 result is mainly because of the acronyms not appearing near to their abbreviation or being not present at all in the corpus. The entire acronym similarity validation data is made available as Supplemental Material (Kumar and Starly 2021).

Table 3 Word embedding training pertinent parameters for word2vec model

Parameter name	Value	Explanation about parameter
Sentences		Tokens of sentences are given as input. Tokenization was done using NLTK
Size	300	This represents dimension of the vectors
Window	10	This is the distance between predicted and current word in a sentence
Minimum count	5	This is a threshold below which all words will be neglected during training of embeddings
Workers	4	This is mainly used to split the processing between cores while training of model, for faster processing
SG	1	Keeping this as '1' enables skip-gram, else utilizes CBOW
Negative	10	It is usually kept between 5–20. Zero negative sampling means no negative sampling has been used
Alpha	0.01	This is the learning rate. Higher learning rate could cause the convergence in the model quickly towards suboptimal solution, and lower could lead the process to stuck. It is one of the most important hyperparameters that should be critically optimized
Sample	10^{-4}	This is the subsampling threshold for higher frequency words
Iter	20	This is the training epochs or the number of iterations throughout the corpus

Table 4 Word embedding similarity results validation

Results	Entities found	%	Percentile
Top	31	44.29	44.29
Top 5	21	30.00	74.29
Top 10	8	11.43	85.72
Top 15	5	7.14	92.86
Not found	5	7.14	

Context of words in a sentence

For any sentence having n words, where words are represented in the form of word embedding, the input into the model is given as the sequence of words, and the algorithm is trained by considering the local context of every word. In this work, long short-term memory networks (LSTMs) were adopted. LSTMs incorporates a memory cell that deals with the long-term dependencies issue. Bi-directional LSTM (BiLSTM), a variant of LSTM, computes the representation of sentence context from forward as well as backward direction. In other words, two layers of LSTMs works in opposite direction, where one reads the sentence in forward direction, while the other in the backward direction. The results are the concatenation of left–right representation of context. The representation of the model is shown in the Fig. 3.

Character level embedding

To get the shape of the word, character-level features are introduced which enables generation of word embedding of a word through the characters. Every character having its own corresponding embedding is used in a similar fashion as words are used for sentences. Character embeddings were learnt during model's training, unlike word embeddings which were pretrained on a custom manufacturing corpus. As like word-BiLSTM, the character embeddings are used in forward and backward propagation through BiLSTM, which is further concatenated to predict a word. For example, Ti6Al4V is a type of titanium alloy that has a feature consisting of characters and numerals in a sequence, and it is recognized as 'material' category. For the output layer of the neural network model, we have employed conditional random fields (CRF) instead of the common softmax layer. CRF outperforms softmax in capturing and producing the valid sequence of output labels and their interdependencies (Lample et al., 2016). Various hyperparameters such as learning rate, learning rate method, learning rate decay, batch size, dropout, number of epochs affects the result of the model. To obtain good performance of NER model hyperparameters are needed to be optimized by training the model using random values (Bergstra & Bengio, 2012). The final hyperparameters, as mentioned in Table 5, were selected based on highest accuracy obtained during the development and testing set.

Results

Precision-recall for NER performance

Upon training the NER model on the corpus of manufacturing literature, named entity extraction was performed on unlabeled manufacturing text, and the model was able to perform information extraction accurately. In Fig. 3, it is shown that a text from the manufacturing literature has been taken and ran through the NER model. Figure 4 shows a representation of the NER classifier results, which demonstrates that it has correctly predicted named entities in nearly every category. The assessment of precision, recall and f1 score are key indicators for measurement of performance in a quantitative manner and the expressions for each one

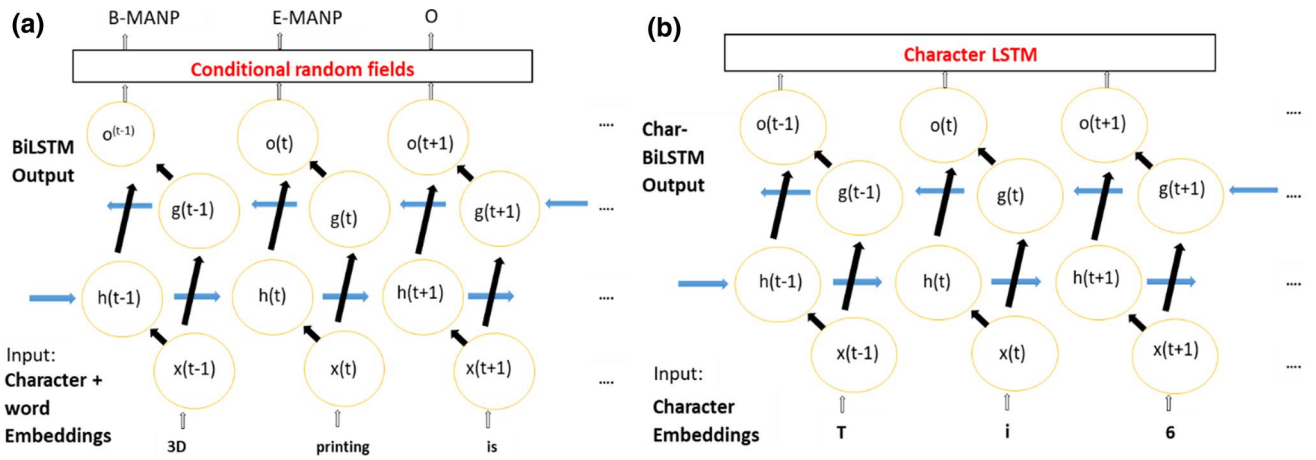

Fig. 3 The architecture of neural network model for named entity recognition in manufacturing text. The architecture in **A** shows word level bidirectional LSTM that is fed with sequence of words (in the form of word embeddings) which returns the tag of respective entity in BIOES format. The word level embedding feature is concatenated with the character level LSTM output for the same word. **B** Shows the character-level LSTM architecture

Table 5 Final hyperparameter values for training NER model

Parameters	Values
Number of epochs	30
Dropout	0.5
Batch size	32
Learning method	Adam
Learning rate	0.01
Learning decay	0.9

of them are given in Fig. 5. For NER, the trained models were evaluated on test data using f1 score, and this evaluation is done for each entity category as shown in Fig. 5. The NER model performance for Manufacturing text is quite decent with an f1 score of 88% which is near to the metrics proposed by state-of-the-art work by Lampe et al. having f1 score closer to 91%. Most importantly, the performance of NER for domain specific text ought to be different as compared to a regular text (Eltyeb & Salim, 2014). As per our current knowledge, there is no manufacturing named entity recognition model developed so far, hence, no comparison could be made on a same scale with any existing models.

The f1 score of MATE (Material) category is highest i.e. 93% while that of MANS (Manufacturing Standard) is lowest i.e. 52%. The high score of Material is most likely due to the fact that the text that was taken randomly for training the model might have greater frequency of occurrence in training set, and due to material entities being mostly 'single tokens' (Weston et al., 2019). Similarly, the low f1 score for the MANS category could be explained by the MANS

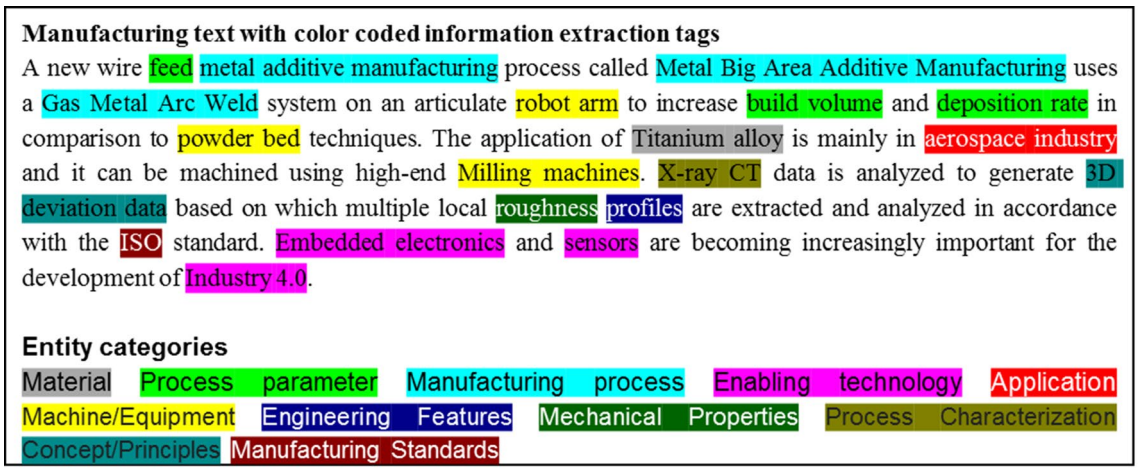

Fig. 4 An example of entity recognition for an unseen manufacturing text

(a)

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$F1\ score = \frac{2 * Precision * Recall}{Precision + Recall}$$

(b)

Category	precision	recall	f1-score
APPL	0.84	0.8	0.82
BIOP	0.88	0.38	0.53
CHAR	0.88	0.81	0.84
CONPRI	0.94	0.87	0.9
ENAT	0.95	0.4	0.57
FEAT	0.91	0.8	0.85
MACEQ	0.95	0.71	0.82
MANP	0.9	0.83	0.87
MANS	0.92	0.35	0.52
MATE	0.97	0.9	0.93
PARA	0.91	0.87	0.89
PRO	0.93	0.82	0.87
Overall	**0.93**	**0.84**	**0.88**

Fig. 5 **a** Expressions for precision, recall and F1 score calculation. **b** Results obtained for our NER model

category having fewer labeled entities, and there could be some labeled entities in the test set that do not exist in the training set. The NER model can then be used to predict the category for all relevant vocabulary words available within the entire corpus of the 500K + abstracts. If the word or group of words was categorized earlier then the priority will be given to manual categorization, and if the word is new and predicted by the model the new category for the word is stored.

Related keywords under various categories for a given query term

One of the applications of NER for Manufacturing could be seen in the form of information tool. A new researcher or someone who is not well versed with the manufacturing process domain might have a hard time browsing the terms or entities and understanding the relation between the entities that appears near to the recognized entity. This is where this tool could be very beneficial to the users who want to either dive into this domain or want to know about the terms for getting a service from manufacturing marketplace. This summary will provide users with information on manufacturing process, application, property etc. and could be very beneficial to non-domain experts for getting acquainted with initial information about the field or for domain experts for obtaining further quantitative information.

As discussed in the earlier section, words appearing near to each other has a cosine distance closer to 1, the application of which could be in the form of identification of relation between subject word with words in every named entity categories. In one example shown in Table 6, related applications that are closely associated with the word "Maraging Steel" is highlighted. The corresponding score for each retrieved 'Application' entity is the cosine distance word similarity. In a more detailed example, the word, SLM (Selective laser melting) is a *manufacturing process* that is known for its *application* in additive manufactured parts. All related words in respective categories are obtained as shown in Table 7. Some of the *equipment* used for SLM are powder bed, laser machine and powder feedstock, while some related *materials* used in SLM process are alloys of aluminum and titanium such as alsi10mg and ti-6al-4v. The entity in question is compared to entities in the respective category using

Table 6 Results showing the query term; "Maraging Steel" and the closely associated entities within the Application category and their corresponding scores

Search term: maraging steel	
Application	Score
Additive manufactured part	0.535
Thin walled components	0.531
Naval applications	0.522
Aircraft component	0.503
Braze welding	0.498
Phenix systems	0.497
Coated abrasives	0.488
Coated carbides	0.477

cosine word similarity, and then the words in the category are sorted in descending order based on scores. This application is dependent on the vocabulary and the respective categories predicted and stored as discussed in the previous section. It has been noted that some words that are "more related" to the word in question may receive a lower score than other words in the category. This could be explained based on the literature text that has been used for developing the word embedding model which has been developed through an unsupervised learning approach without human intervention. In every case, the most relevant words would be visible to the user for desired target word.

To obtain the similar words related to entity, word embeddings developed earlier did not work, and we needed to develop another word embedding. The reason behind this is the fact that the word and character embedding trained earlier can identify the label as the output but not the similar words that are n-grams. For example, previous word2vec model could find out the similar words related to 'manufacturing' but not related to 'additive manufacturing' as unigram word embedding for BiLSTM was needed for training the sequence of words. However, there could be many n-gram entities that is in the vocabulary of manufacturing. To counter this, we trained another word embedding after

Table 7 Summary about target word 'slm'

Application	Score	Enabling tech	Score	Property	Score	Manufacturing process	Score
additive_manufactured_part	0.524	laser_scan	0.531	de-powdering	0.533	selective_laser_melting	0.780
aircraft_component	0.488	additive_technology	0.531	porosity_density	0.475	l-pbf	0.736
ti_implant	0.468	laser	0.490	precipitated_hardened	0.472	ebm	0.704
bone_implant_applications	0.464	electron_beam-based	0.463	homogeneous_microstructures	0.461	powder_bed_fusion	0.686
thin-walled_components	0.462	atomized	0.435	handling_strength	0.459	sebm	0.683
dental_application	0.460	hybrid_technologies	0.435	youngmodulus	0.452	slmed	0.680
phenix_systems	0.460	additive_technologies	0.433	martensitic_grade	0.442	dmls	0.668
eos	0.458	powder_technologies	0.427	green_part	0.437	selective_laser_melting_process	0.664
Concept principle	Score	Feature	Score	Machine equipment	Score	Manufacturing standard	Score
scanning_strategies	0.564	micro-lattice_structures	0.510	powder_bed	0.604	cad_file	0.391
build_strategy	0.540	functionally_graded_lattice	0.506	am_part	0.551	iso/astm	0.391
geometrically-complex	0.520	lattice_design	0.504	direct_metal_laser	0.531	iso_25178-2	0.365
rapid_solidification_process	0.516	graded_microstructure	0.486	powder_feedstock	0.524	iso_standard	0.361
wrought_samples	0.506	overhang_features	0.481	building_platform	0.519	obj	0.346
samples_manufactured	0.503	lattice_structure_design	0.476	powder_beds	0.505	text_file	0.340
melt_pool_boundaries	0.501	overhanging_feature	0.475	l-pbf_systems	0.482	class_iii_defect	0.316
post-processing_parameters	0.501	mg_scaffolds	0.468	build_plate	0.471	stl_format	0.296
Characterization	Score	Parameter	Score	Material	Score		
build_rates	0.568	hatch_spacing	0.550	alsi10mg	0.688		
building_of_parts	0.492	laser_energy_density	0.548	alsi10mg_alloy	0.609		
mechanical_property_characterization	0.489	build_height	0.536	ti-6al-4v_powder	0.582		
dental_crowns	0.482	scanning_speed	0.534	ti6al4v	0.576		
degrees_of_porosity	0.473	hatch_distance	0.530	in718	0.567		
meltpool	0.471	building_direction	0.519	alsi10mg_alloys	0.566		
three-point_bending_fatigue_tests	0.468	scan_pattern	0.518	co-cr_dental_alloy	0.561		
deposition_quality	0.467	melt_pool_dimension	0.504	aluminium_alloy_alsi10mg	0.557		

replacing all n-gram words by removing spaces between them and placing an underscore to make them act as unigram instead of n-grams. For instance, 'metal additive manufacturing' was replaced with 'metal_additive_manufacturing', and once trained following the process of word embedding would give the results similar to metal additive manufacturing based on appearance of this word in all 12 categories as per the context such that 'Application' category could indicate aerospace as the closest to aerospace out of all applications. A threshold of 0.5 cosine similarity was applied to avoid words not very relevant appearing as similar words.

Topic category identification of a manufacturing text paragraph

Another application of the NER model is for the algorithm to conduct topic assessment exercises based on a given paragraph. One of the most common methods of topics identification is using Latent Dirichlet allocation (LDA) model (Blei et al., 2003). The limitation of LDA is the fixed number of topics that should be known ahead. We tried to make the task of topic assessment simpler by using the NER model to find the focused named entities. Focused named entities (Zhang et al., 2004), are relevant entities mainly concerned with 'Who' and 'What', are vital for identifying the main topic of the paragraph content. Some of the other applications of focused named entities include text summarization, search ranking, topic tracking and topic detection. Towards this portion of study, we focused on topic identification. The topic was chosen to assess a given paragraph text by finding the maximum number of named entities occurring in the paragraph. In case of two or more entities present in equal numbers, the preference will be given to the first identified entity of the paragraph. This is because in most of the cases the sentence starts with introduction about the topic in question. To demonstrate this, two example paragraph texts are shown in Figure 6. The first paragraph discusses about STL file topic. Initially the entities were recognized by NER model and based on the number of occurrences of specific entities, the topic was finalized along with NER category. In the second example, even though '3DP' and 'print' are the two topics identified, the preference was given to the first identified topic at the beginning of the article.

Discussion

Developing an information and relationship extraction method from free flowing manufacturing literature are critical steps to mining the vast literature available on product design and manufacturing text. Future applications can include powering chat bots specific to the design and manufacturing domain, understanding text written in maintenance records and the general parsing of documentation in product manufacturing domain.

Currently, none of the existing NLP models is able to understand manufacturing domain specific text, primarily due to the lack of datasets and more importantly limited data available to train special purpose TLP. While there have been advances in NLP techniques as applied to the biomedical and material science domain, those models do not easily translate to the product engineering lifecycle context. With the advent of online resources, the textual information written in the documentation of manufacturing sector companies, is either publicly available or privately available within the organization. Such digitally available textual content would need to be parsed for retrieval based on a query search. Within the manufacturing sector, there are several kinds of product manufacturing text that is relevant, ranging from product design data, engineering test data, supplier data, maintenance records and product use

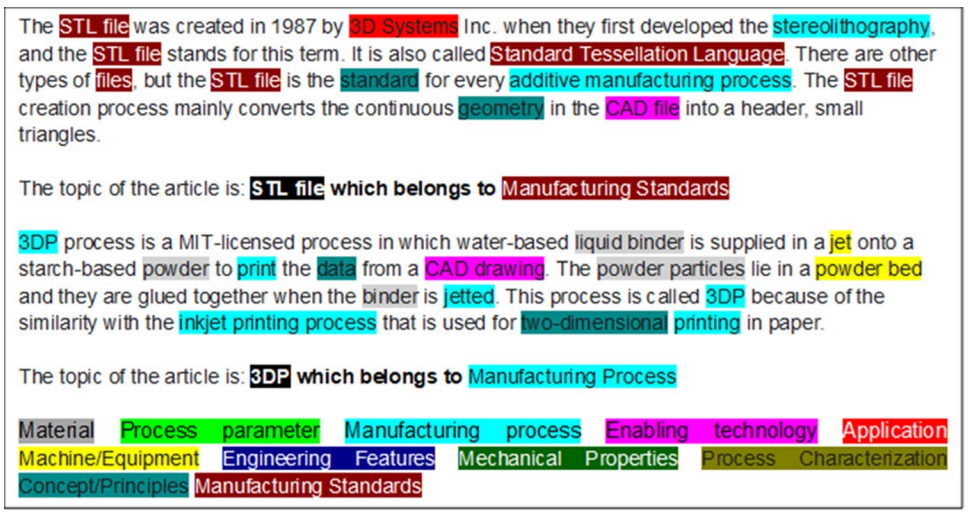

Fig. 6 NER and topic identification of unseen article

data. Often times, such data is not entirely text based, but a combination of images, 3D models, graphs, video and audio content. For automated mining of such domain specific data, building neural network models that are able to classify text is critical to inter-linking various data assets, which eventually lead to information extraction and knowledge based generation.

This work demonstrates a process through which a large amount of manufacturing text abstracts can be mined for analysis and entity recognition. The approach taken here is a semi-supervised approach which uses a combination of human annotation, automatic vocabulary extraction and model based entity categorization. The method of developing a vocabulary to tag the entities is time consuming but quite beneficial for performing the same operations. There are many entities present in the corpus that has variations in their usage in different sentences. These type of variations in the corpus would require entity normalization (Cho et al., 2017) of every entity such that interchangeability of the terms does not affect the embedding and their nearby elements. Although the method discussed earlier based on Levenshtein distance was able to catch most of the variations in entities taken for training, there is a possibility that variation in terms could have been left out. The approach also intentionally chose to lowercase every term while automatically tagging based on vocabulary to avoid complexities arriving due to sentence formation, some entities such as Acrylonitrile Styrene (AS) having the lowercase acronym 'as' overlaps with the adverb 'as', which can lead to errors within the model. In such situations, manual checking or applying conditions for checking common words with abbreviations would have to be resorted to rectify ambiguous entities.

Some of the results obtained through this approach might seem incomplete since we are only analyzing the words contained within the abstracts. Analysis of full text associated with each article may provide a more complete picture of the entities within the domain. Also, including content from non-academic work would also be beneficial since it would add words/entities that might help interlink into other related domains. But in any case, the presented approach is still valid and would simply be an issue of scaling the model to also include other forms of full-form text. Adding vast amounts of additional data would also require further fine tuning of the models through hyper-parameter optimization techniques, such as grid search, and training data to help improve classification accuracy.

The word2vec technique is effective for named entity recognition but there are newer techniques such as BERT and Attention based algorithms that have gained prominence over the last few years (Zhang et al., 2021). In an effort to replace word2vec models, a custom pre-trained model meant for scientific text, called 'SciBERT' is available (Beltagy et al., 2019). Such models can be fine-tuned to fit specific manufacturing text corpora. Fine-tuning of SciBERT with a relatively smaller dataset would be quite beneficial in improving the performance in many NLP tasks such as sequence tagging (leading to NER), sentence classification, dependency parsing and question-answering (QA) systems. Another area of improvement is the inter-linking of various concepts available within the literature to form a Knowledge network graph that connects the various entities together (Shen et al., 2014) within sub-disciplines or across disciplines. Such manufacturing specific knowledge graphs can further enhance information extraction and information retrieval based on query terms.

Conclusion

This work demonstrates an approach to entity recognition from more than half a million manufacturing related scientific text with quite good accuracy. The outlined approach can be used to perform the named entity recognition for any domain-specific text related to manufacturing, such as manufacturing maintenance records, service manuals or other business specific documents. We have created the word embedding model for discrete manufacturing using the scientific text corpus and achieved more than 74 percentile results in the top 5 most similar results. We have developed a named entity recognition model for Manufacturing NER, and achieved closer to the state-of-the-art performance i.e., accuracy equivalent to 88% using BiLSTM and CRF based neural network model. We have relied on an unsupervised learning approach to create the word-embeedings and a supervised approach to training the BiLSTM network. To further extend the use of annotated datasets while reducing labor expense, methods on semi-supervised learning, continuous learning, utilizing structured content available in databases and reference materials could be possible approaches to create additional labelled datasets without relying entirely on human annotation. Two useful applications of NER has been demonstrated in the form of literature summary review and article topic identification. This work represents the initial step towards building knowledge network graphs within the entire manufacturing domain that can help lower the barriers towards access to information by both domain and non-domain experts.

Acknowledgements We thank support from the NSF Grant#1937043 for funds to carry out portion of this work.

Funding Funding was provided by Directorate for Engineering.

References

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon, 4*(11), e00938.

Acronyms similarity data https://doi.org/10.6084/m9.figshare.14785266.

Alfeo, A. L., Cimino, M. G., & Vaglini, G. (2021). Technological troubleshooting based on sentence embedding with deep transformers. *Journal of Intelligent Manufacturing, 7*, 1–2.

Ali, N. Chatbot: A conversational agent employed with named entity recognition model using artificial neural network. arXiv preprint arXiv:2007.04248. 2020 Jun 19.

Beltagy, I., Lo, K., & Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. arXiv preprint arXiv:1903.10676.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research 13*(2).

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research, 3*, 993–1022.

Brewer, A., Sloan, N., & Landers, T. L. (1999). Intelligent tracking in manufacturing. *Journal of Intelligent Manufacturing., 10*(3), 245–250.

Brundage, M. P., Sexton, T., Hodkiewicz, M., Dima, A., & Lukens, S. (2021). Technical language processing: Unlocking maintenance knowledge. *Manufacturing Letters, 1*(27), 42–46.

Chen, Z., Liu, X., Yin, Y., & Lu, H. (2020). Named entity recognition method for fault knowledge based on deep learning. In *Proceedings of the 4th international conference on machine learning and soft computing* (pp. 1–4).

Cho, H., Choi, W., & Lee, H. (2017). A method for named entity normalization in biomedical articles: Application to diseases and plants. *BMC Bioinformatics, 18*(1), 1–2.

Costa, R., Lima, C., Sarraipa, J., & Jardim-Gonçalves, R. (2016). Facilitating knowledge sharing and reuse in building and construction domain: An ontology-based approach. *Journal of Intelligent Manufacturing, 27*(1), 263–282.

Cui, W., Xiao, Y., Wang, H., Song, Y., Hwang, S. W., & Wang, W. (2019). KBQA: learning question answering over QA corpora and knowledge bases. arXiv preprint arXiv:1903.02419.

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

Eltyeb, S., & Salim, N. (2014). Chemical named entities recognition: A review on approaches and applications. *Journal of Cheminformatics, 6*(1), 1–2.

Gasmi, H., Bouras, A., & Laval, J. (2018). LSTM recurrent neural networks for cybersecurity named entity recognition. *ICSEA, 14*(11), 2018.

Goldberg, Y., & Levy, O. (2014). word2vec Explained: Deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722.

Gusenbauer, M. (2019). Google Scholar to overshadow them all? Comparing the sizes of 12 academic search engines and bibliographic databases. *Scientometrics, 118*(1), 177–214.

Haldar, R., & Mukhopadhyay, D. (2011). Levenshtein distance technique in dictionary lookup methods: An improved approach. arXiv preprint arXiv:1101.1232.

Honnibal, M., & Johnson, M. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1373–1378).

Kassner, L., Hirmer, P., Wieland, M., Steimle, F., Königsberger, J., & Mitschang, B. (2017). The social factory: Connecting people, machines and data in manufacturing for context-aware exception escalation. In *Proceedings of the 50th Hawaii international conference on system sciences*.

Kejriwal, M. (2019). *Domain-specific knowledge graph construction*. Springer International Publishing.

Kumar, A., Alam, Z., Khan, D. A., & Jha, S. (2019). Nanofinishing of FDM-fabricated components using ball end magnetorheological finishing process. *Materials and Manufacturing Processes, 34*(2), 232–242.

Kumar, A., & Starly, B. (2021). Dataset_NER_Manufacturing—"FabNER": Information Extraction from Manufacturing Process Science Domain Literature Using Named Entity Recognition. figshare. Dataset. https://doi.org/10.6084/m9.figshare.14782407.v1.

Kung, H. K., Hsieh, C. M., Ho, C. Y., Tsai, Y. C., Chan, H. Y., & Tsai, M. H. (2020). Data-augmented hybrid named entity recognition for disaster management by transfer learning. *Applied Sciences, 10*(12), 4234.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360.

Leaman, R., & Gonzalez, G. (2008). BANNER: An executable survey of advances in biomedical named entity recognition. *InBiocomputing, 2008*, 652–663.

Lende, S. P., & Raghuwanshi, M. M. (2016). Question answering system on education acts using NLP techniques. In: *2016 world conference on futuristic trends in research and innovation for social welfare (Startup Conclave)* (pp. 1–6). IEEE.

Levy, O., & Goldberg, Y. (2014). Linguistic regularities in sparse and explicit word representations. In *Proceedings of the eighteenth conference on computational natural language learning* (pp. 171–180).

Li, J., Sun, A., Han, J., & Li, C. (2020). A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*.

Li, L., Wang, P., Yan, J., Wang, Y., Li, S., Jiang, J., Sun, Z., Tang, B., Chang, T. H., Wang, S., & Liu, Y. (2020b). Real-world data medical knowledge graph: construction and applications. *Artificial Intelligence in Medicine, 103*, 101817.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: System demonstrations* (pp. 55–60).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Mittal, V., Mehta, P., Relan, D., & Gabrani, G. (2020). Methodology for resume parsing and job domain prediction. *Journal of Statistics and Management Systems, 23*(7), 1265–1274.

Nadeau, D., & Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes, 30*(1), 3–26.

Navinchandran, M., Sharp, M. E., Brundage, M. P., & Sexton, T. B. (2021). Discovering critical KPI factors from natural language in maintenance work orders. *Journal of Intelligent Manufacturing, 22*, 1–9.

Nguyen, D. Q., Zhai, Z., Yoshikawa, H., Fang, B., Druckenbrodt, C., Thorne, C., Hoessel, R., Akhondi, S. A., Cohn, T., Baldwin, T., & Verspoor, K. (2020). ChEMU: Named entity recognition and event extraction of chemical reactions from patents. In *European conference on information retrieval 2020 Apr 14* (pp. 572–579). Springer, Cham.

Oruç, O., & Aßmann, U. (2020). A semantic question answering in the domain of smart factories. EasyChair.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543).

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

Ratinov, L., & Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the thirteenth conference on computational natural language learning (CoNLL-2009)* (pp. 147–155).

Řehůřek, R., & Sojka, P. (2011). Gensim-statistical semantics in python. Retrieved from genism.org.

Rocktäschel, T., Weidlich, M., & Leser, U. (2012). ChemSpot: A hybrid system for chemical named entity recognition. *Bioinformatics, 28*(12), 1633–1640.

Shah, N., Engineer, S., Bhagat, N., Chauhan, H., & Shah, M. (2020). Research trends on the usage of machine learning and artificial intelligence in advertising. *Augmented Human Research., 5*(1), 1–5.

Shahzad, K. M., & Hadj-Hamou, K. (2013). Integrated supply chain and product family architecture under highly customized demand. *Journal of Intelligent Manufacturing., 24*(5), 1005–1018.

Sharma, A., Adhikary, A., & Borah, S. B. (2020). Covid-19′ s impact on supply chain decisions: Strategic insights from NASDAQ 100 firms using Twitter data. *Journal of Business Research, 1*(117), 443–449.

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

Stewart, M., Enkhsaikhan, M., & Liu, W. (2019). Icdm 2019 knowledge graph contest: Team uwa. In *2019 IEEE international conference on data mining (ICDM)* (pp. 1546–1551). IEEE.

Studiawan, H., Sohel, F., & Payne, C. (2018). Automatic log parser to support forensic analysis. 2018. In *Conference: 16th Australian digital forensics conference at: Edith Cowan University, Perth, Australia.*

Tshitoyan, V., Dagdelen, J., Weston, L., Dunn, A., Rong, Z., Kononova, O., Persson, K. A., Ceder, G., & Jain, A. (2019). Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature, 571*(7763), 95–98.

Weston, L., Tshitoyan, V., Dagdelen, J., Kononova, O., Trewartha, A., Persson, K. A., Ceder, G., & Jain, A. (2019). Named entity recognition and normalization applied to large-scale information extraction from the materials science literature. *Journal of Chemical Information and Modeling, 59*(9), 3692–3702.

Yan, H., Yang, J., & Wan, J. (2020). KnowIME: A system to construct a knowledge graph for intelligent manufacturing equipment. *IEEE Access, 28*(8), 41805–41813.

Zhang, L., Yue P., & Tong Z. (2004). Focused named entity recognition using machine learning. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 281–288).

Zhang, W., Dong, C., Yin, J., & Wang, J. (2021). Attentive representation learning with adversarial training for short text clustering. *IEEE Transactions on Knowledge and Data Engineering*.

Zheng, S., Hao, Y., Lu, D., Bao, H., Xu, J., Hao, H., & Xu, B. (2017). Joint entity and relation extraction based on a hybrid neural network. *Neurocomputing, 27*(257), 59–66.

Zuzarte F, Proença M. Cloud services in supply chains (Doctoral dissertation). 2019.