Discovering Content through Text Mining for a Synthetic Biology Knowledge System

Bridget T. McInnes[1],* J. Stephen Downie[2], Yikai Hao[4], Jacob Jett[2], Kevin Keating[3], Gaurav Nakum[4], Sudhanshu Ranjan[4], Nicholas E. Rodriguez[1], Jiawei Tang[4], Du Xiang[4], Eric M. Young[3], and Mai H. Nguyen[4]

[1] *Virginia Commonwealth University, Richmond VA 23284, USA,* [2] *University of Illinois at Urbana-Champaign, Urbana IL 61801, USA,* [3] *Worcester Polytechnic Institute, Worcester MA 01609, USA,* [4] *University of California San Diego, La Jolla CA 92093, USA*

E-mail: btmcinnes@vcu.edu

Abstract

Scientific articles contain a wealth of information about experimental methods and results describing biological designs. Due to its unstructured nature and multiple sources of ambiguity and variability, extracting this information from text is a difficult task. In this paper, we describe the development of the SBKS text processing pipeline. The pipeline uses natural language processing techniques to extract and correlate information from the literature for synthetic biology researchers. Specifically, we apply named entity recognition, relation extraction, concept grounding, and topic modeling to extract information from published literature in order to link articles to elements within our knowledge system. Our results show the efficacy of each of the components on synthetic biology literature, and provide future directions for further advancement of the pipeline. Keywords: Synthetic Biology Text Processing Pipeline, Natural Language Processing, Named Entity Recognition, Relation Extraction, Concept Grounding, Topic Modeling

Introduction

The field of synthetic biology has seen exciting growth in the last few years. These articles contain a wealth of information about experimental methods and results on biological designs. Although the amount of data and publications has increased tremendously with numerous available data sources that are fragmented, making it challenging to locate relevant data for genetic design. For example, finding sequence and performance data for biological parts remains a manual process of sifting through articles and supplemental material. To address this, we are developing our Synthetic Biology Knowledge System (SBKS) that integrates disparate data and publication repositories to deliver effective and efficient access to available information.[1]

Named entities in biological literature, such as genes, proteins, chemicals, and cell lines, are often referenced using ambiguous symbols.[2] For example, abbreviations like GAP can have a different semantic meaning depending on the context, like glyceraldehyde 3-phosphate in glycolysis, GTPase-activating protein in cell cycle regulation, or its common meaning in the English language. Furthermore, the meaning and synonym use of named entities can vary across organisms, documents, and author networks.[3] In synthetic biology, cell lines are often given names that are a concatenation of the species name and the genes that were modified within their genome, leading to nested annotations of mixed types. These and other sources of ambiguity and variability create challenges for extracting and associating information from biological text which necessitate the use of natural language processing methods.

This paper describes the development of a text processing pipeline using natural language processing (NLP) techniques to extract and correlate information from the literature for synthetic biology researchers. Namely, we apply XML parsing, Named Entity Recognition (NER), Relation Extraction (RE), Concept Grounding, and Topic Modeling to identify concepts and identities in published articles in order to link each article to other elements in our knowledge system.

The paper is structured as follows. First, we describe the background and related work for each of the components in our pipeline. Second, we describe the data used to train and evaluate our components. Third, we describe the methods for each component. Fourth, we discuss the results achieved with our pipeline. Finally, we describe the future directions of the pipeline.

Background and Related Work

In this section, we describe at a high level each component and their related works.

Named Entity Recognition

The goal of Named Entity Recognition (NER) is to locate and classify named entities present in text into pre-defined categories.[4] For synthetic biology, examples of such categories are names of genes, vectors, and regulatory elements. NER in biology domains has additional challenges due to the pace of new named entities being added, lack of naming convention, lengthy names, presence of special characters, and frequent and variable use of abbreviations.[5,6] Figure 1 shows an example sentence containing Chemicals, Species and Cell line entities.

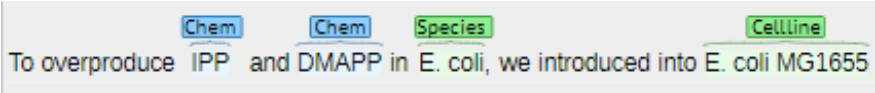

Figure 1: Example sentence containing mentions of Chemicals, Species and Cell Line entities.

Deep neural network approaches have been applied to NER on biomedical texts. Specifically, state-of-the-art approaches use Long Short-Term Memory (LSTM)[7] with Conditional Random Field (CRF)[8] models and Transformers.[9] For example, the Bidirectional Encoder Representations from Transformers (BERT), a neural network-based language model that produces contextualized word embeddings which are typically fed into a neural network top model, has been shown to perform well across a variety of NLP tasks,[10] including NER.[11]

BioBERT extends the BERT language model to biological domains by supplementing the pre-training data with PubMed Abstracts and PMC full-text articles.[12]

In our work, we fine-tune BioBERT pre-trained models on each NER category separately. Linking the results of NER processes to concepts through concept grounding is a more recent approach that has been taken first in biomedical domains[13] and then more broadly to information problems in general.[14]

Relation Extraction

The goal of Relation Extraction (RE) is to automatically determine if there is a relationship between two entities and if so, classify the type of relation between them. For synthetic biology, example relations include UpRegulator, DownRegulator, and Subtrate, as shown in Figure 2. In the example shown in the figure, three instances of the relation Subtrate are identified (e.g., the entities *mevalonate pathway* and *acetyl-CoA* have the relationship of Subtrate).

Figure 2 shows an example sentence containing a DownRegulator relationship between Chemical and Gene entities.

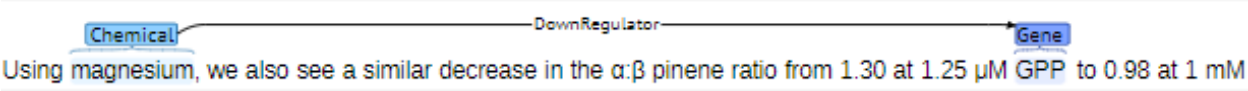

Figure 2: Downregulator relation identified by the Relation Extraction model

Deep neural networks have also been successfully used for this task. Earlier approaches used variants of Convolutional Neural Networks (CNN)[15] or Long Short-Term Memory (LSTM)[7] or a mixture of both. Some methods provide extra context on how words are related to each other by passing part-of-speech (POS) tags to the model. As with NER, more recent approaches use Transformers[9] and their variants. Zhang et al.[16] give an extensive review of neural network-based approaches for biomedical relation extraction. Similar to the NER task, we use BioBERT for relation extraction. We also use another neural net-

work as a top model to further process BioBERT embeddings for relation classification to construct the final RE model.

Concept Grounding

The goal of Concept Grounding (also referred to as entity linking[17] or normalization[18]) is to map entity mentions to their concepts in their respective ontologies or taxonomies.[19] For example, synonymous labels such as *Escherichia coli*, *E. coli*, *E. coil*, *e. coli*, *e. coil*, etc. are normalized to their unique identifier *NCBI:txid562*, which references the concept for the *Escherichia coli* organism in the National Center for Biotechnology Information's (NCBI)[20] organism taxonomy. This is a challenging task as these mentions come from various entity types, have a wide range of lexical diversity, and whose concepts often exist across a set of different taxonomies.[21] Historically, the automated concept grounding of terms to concepts in an ontology were rule-based systems. Systems such as MetaMap[22–24] automatically map terms in biomedical text to concepts in the Unified Medical Language System (UMLS) using a series of linguistic rules and patterns. It is one of the first automated systems for concept grounding, and is the backbone of Medical Text Indexer (MTI),[25] a system to identify Medical Subject Heading terms in PubMed abstracts to aid the indexing process at the National Library of Medicine, National Institutes of Health. Recently, due to the number of corpora that have been developed to evaluate concept grounding systems, researchers have focused on machine learning. These systems can be divided into two categories: those that treat the problem as a classification task,[26–30] and those that treat the problem as a rank prediction task.[31–39] For the classification category, researchers have been utilizing neural networks, and although these have been working well with smaller datasets, there still is a difficulty when scaling to larger ontologies. One reason is that the output space must be the same size as the number of concepts to be predicted, and therefore only works well when the output space tends to be small.[40] For the rank prediction category, the algorithm transforms input representations of text mentions to optimize the similarity between the

final transformed vectors of mentions and the vectors of their associated concepts in order to identify the appropriate concept.[31]

In a separate work, we are currently developing a rank prediction system, and are developing a training data set specifically for synthetic biology concept grounding and analyzing which ontologies are required to be integrated into our system.

Topic Modeling

The goal of topic modeling is to uncover latent semantic structures in a collection of text documents. Using a probabilistic approach, topic modeling represents a topic as a cluster of similar words, and a document as a set of topics.[41] Using the statistics of the words in a corpus, topic modeling uncovers topics based on words that frequently occur together, and determines the set of topics and their proportions in each document.

There are several approaches to topic modeling. In this work, we use Latent Dirichlet Allocation (LDA).[42] LDA has been used to extract information from literature in many scientific disciplines, including bio-related fields.[43] Some groups have extended LDA with bio-related terms to find complex biological relationships in PubMed articles.[44]

Data

Training Data

In this section, we describe the data used to develop the components in our text mining pipeline.

HUNER Data

For training and evaluating our NER system, we utilize the HUNER dataset,[45] which consists of 34 different corpora covering four entity types: Chemicals, Cell Lines, Genes/Proteins, and Species. The data is partitioned into 60% training, 10% validation, and 30% testing.

Table 1: The mention counts in the HUNER development, test and training data sets.

Entity	Dataset		Total			Unique		
		Type	DEV	TEST	TRAIN	DEV	TEST	TRAIN
CELL LINE	**CLL**	AA	30	77	234	26	67	195
	GELLUS	AA	75	247	328	32	99	110
	JNLPBA	AA	429	1117	2284	286	771	1383
CHEMICAL	**CDR**	AA	1511	4716	9207	560	1503	2461
	CEMP	PF	6364	18958	39293	3093	7506	14240
	CHEBI	PF	1262	6067	8779	594	2077	2627
	CHEMDNER	AA	8062	24288	48347	3687	9035	16094
GENE	**BC2GM**	AA	2163	6753	14456	1938	5513	10846
	BIOINFER	AA	455	1383	2658	244	597	987
	DECA	AA	576	1776	3670	250	772	1457
	FSU	AA	6606	19383	33505	2539	6429	9878
	GPRO	PA	1315	3576	7832	900	2004	3958
	IEPA	AA	104	300	708	46	81	146
	JNLPBA	AA	3029	8777	18463	1306	3195	6029
	MIRNA	AA	76	291	541	38	129	234
	OSIRIS	AA	96	291	535	34	114	234
	VARIOME	AF	300	1082	3045	65	214	509
SPECIES	**LINNEAUS**	AF	85	278	566	26	41	70
	MIRNA	AA	64	227	385	12	34	31
	S800	AA	406	1074	2188	203	518	1044
	VARIOME	AF	33	66	83	3	7	6

In this work, we evaluate 21 of the datasets to determine the efficacy of using pre-existing training data to extract relevant entities from full text synthetic biology articles. Table 1 shows data sources for each entity type, the type of text that is being annotated (Abstract (AA), Patents (PA), Full text article (FA)), the number of mentions (and unique mentions) within the training, test, and development datasets.

ChemProt Data

For training and evaluationg our RE system, we use the ChemProt dataset,[46] which consists of sentences containing chemical-protein interactions extracted from PubMed abstracts and annotated by domain experts. This dataset was used in the BioCreative VI chemical-protein interaction challenge.[17] We use a processed version of the original ChemProt dataset[47] that contains sentences from PubMed abstracts where each sentence contains a pair of entities and a label indicating the relation between the two entities. There are five classes of ChemProt Relation (CPR) in the processed ChemProt dataset. The relations are CPR:3 (UpRegulator),

CPR:4 (DownRegulator), CPR:5 (Agonist), CPR:6 (Antagonist), and CPR:9 (Subtrate). The processed dataset is split into train, development, and test datasets. The size (number of sentences) of the dataset is shown in Table 2. Table 3 shows the distribution of sentences across the relation classes in the dataset.

Table 2: The number of sentences in the ChemProt development test and training datasets

	TRAIN	DEV	TEST
Number of sentences	19,460	16,943	11,820

ACS Data

For evaluating our NER and RE systems, we use the American Chemical Society (ACS) dataset which comprises of full text articles from ACS Synthetic Biology. The data set contains 1,545 articles with supplemental files between the years 2011 and 2019. Two full text articles were randomly extracted and annotated for both entities (Chemicals, Cell Lines, Genes/Proteins, and Species) and their relations (UpRegulator, DownRegulator and Subtrate) by two synthetic biology experts on our team. The other two relations (Agonist and Antagonist) were not annotated in the two ACS articles as these relations are redundant with UpRegulator and DownRegulator in the context of synthetic biology.

Table 3 shows the distribution of sentences across the relation types for the ACS dataset used for testing compared to the ChemProt dataset used for training described above. Since the ACS dataset does not contain any annotations for Agonist and Antagonist, we excluded these relations from our RE process and analysis. The following subsections describe the annotation process for both NER and RE.

NER Annotations

Table 4 shows the confusion matrix of the annotator's label choices. The first row indicates the beginning-inside-outside (BIO) label and entity type for one annotator, and the first

Table 3: Distribution of sentences across relation types in ChemProt and ACS datasets

Relation	ChemProt			ACS
	Train	Dev	Test	Test
UpRegulator (CPR:3)	3.95%	4.65%	3.92%	2.41%
DownRegulator (CPR:4)	11.57%	9.26%	9.80%	4.05%
Agonist (CPR:5)	0.89%	0.98%	1.15%	N/A
Antagonist (CPR:6)	1.21%	1.68%	1.73%	N/A
Subtrate (CPR:9)	3.74%	3.87%	3.80%	14.38%
No Relation	78.65%	79.56%	79.59%	79.14%

column indicates the BIO label and entity type for the other annotator. The BIO format indicates the location of the words within the entities. These labels include B=beginning, I=inside, and O=outside (or not a named-entity of interest). The values in the table represent the number of tokens that were labeled by the annotators. For example, the value in the bottom right corner means 25 tokens were labeled as being inside of a cell line entity by both annotators. The majority of mismatched labels between the two annotators are between Gene-Chemical, and Species-Cell Line.

There are 144 mismatched labels between B-Gene and O (not an entity), and 59 mismatched labels between B-Chemical and O. Many of these are Genes, Proteins, and Chemicals that are commonly used in synthetic biology experiments such as restriction enzymes, but are not related to the biological system on which the article was focused. Consequently, one annotator chose to ignore them.

Table 5 shows NER evaluation metrics of each annotator using the adjudicated data set as the ground truth. The adjudicated data is the consensus from the annotators after discussing their disagreements. The values are grouped by entity type and averaged. An exact match means that the annotator's label and the boundaries of the labeled entity are exactly the same. An approximate match allows the entity boundaries to overlap with the ground truth, i.e., a partial match. Cohen's κ coefficient is also reported and measures the agreement between the annotators and accounts for the underlying probability of agreement for any

token if hypothetically the labels were randomly assigned by chance. When considering all entity types, the annotators obtained a $\kappa = 0.9050$ indicating almost perfect agreement.

Table 4: Inter-annotator agreement confusion matrix of NER labels

	O	B-Gene	I-Gene	B-Chemical	I-Chemical	B-Species	I-Species	B-Cellline	I-Cellline
O	15910	6	8	29	43	3	2	0	0
B-Gene	144	502	2	22	0	0	1	0	0
I-Gene	32	12	142	0	4	0	0	0	0
B-Chemical	59	6	1	804	23	0	0	0	0
I-Chemical	24	0	4	10	517	0	0	0	0
B-Species	0	0	0	0	0	204	0	13	0
I-Species	0	0	0	0	0	0	391	0	26
B-Cellline	2	0	0	0	0	2	0	7	13
I-Cellline	6	2	0	0	0	0	4	0	25

Table 5: Comparison of annotators to the adjudicated NER annotations

		Exact			Approximate		
Annotator	Entity	Precision	Recall	F1	Precison	Recall	F1
1	Cell line	0.8750	0.9545	0.9130	0.9167	1	0.9565
	Chemical	0.9308	0.9145	0.9226	0.9877	0.9672	0.9774
	Gene	0.8904	0.9442	0.9165	0.9108	0.9811	0.9447
	Species	0.9954	0.9686	0.9818	1	0.9731	0.9864
	All	0.9226	0.9323	0.9274	0.9594	0.9733	0.9663
2	Cell line	0.3000	0.2727	0.2857	1	0.9091	0.9524
	Chemical	0.9954	0.9452	0.9696	0.9988	0.9485	0.973
	Gene	0.9906	0.8140	0.8936	0.9943	0.8183	0.8978
	Species	0.9952	0.9372	0.9654	0.9952	0.9372	0.9654
	All	0.9852	0.8890	0.9347	0.9969	0.9001	0.946
							Cohen's $\kappa = 0.9050$

RE Annotations

Table 6 denotes the inter-annotator agreement for RE task on the ACS dataset. We follow the same convention as used in the NER confusion matrix table. We can observe that the annotators agree on all of the labeled relations. The disagreement occurs for the relations that are skipped or marked as No Relation. We can see that a large number of relations marked as DownRegulator (12) or Subtrate (72) by one annotator are marked as No Relation by the other annotator.

Table 7 shows evaluation metrics for each annotator with the adjudicated dataset used as ground truth on the RE task. The values are grouped by relation type and averaged. For the annotator agreement on the RE task, the Cohen's κ coefficient is 0.7442.

Table 6: Inter-annotator agreement confusion matrix of RE labels

	UpRegulator	DownRegulator	Subtrate	NoRelation
UpRegulator	5	0	0	0
DownRegulator	0	17	0	12
Subtrate	0	0	78	72
NoRelation	0	23	53	1121

Table 7: Comparison of annotators to the adjudicated RE annotations

		Annotator 1			Annotator 2		
		Precision	Recall	F1	Precision	Recall	F1
Adjudicated	UpRegulator	1.0000	0.2000	0.3333	1.000	0.2000	0.3333
	DownRegulator	0.5862	0.4047	0.4789	0.8	0.7619	0.7804
	Subtrate	0.5866	0.5906	0.5886	0.6319	0.6107	0.6211
	NoRelation	0.8773	0.9117	0.8942	0.8704	0.9299	0.8991
	All Classes	0.8265	0.8275	0.8196	0.8508	0.6340	0.6746
							Cohen's $\kappa = 0.7442$

Methods

In this section, we describe the methods behind each component of our text mining pipeline, including 1) text parsing, 2) Named Entity Recognition, 3) Relation Extraction, 4) Concept Grounding, and 5) Topic Modeling. Figure 3 provides an overview of each of the different components and how the components are integrated into a single pipeline. The tools developed for the pipeline can be found at `http://web.synbioks.org/` and `https://github.com/synbioks`.

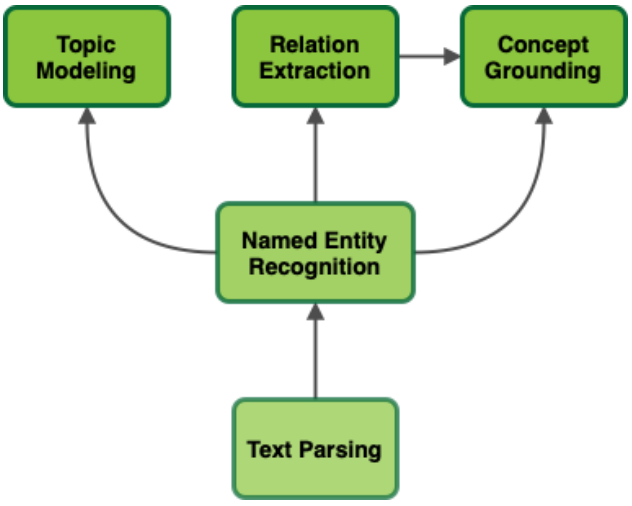

Figure 3: Components in our text mining pipeline

Text Parsing

The articles from ACS are in XML format. We developed a preprocessing step that parses the XML files and extracts metadata and raw texts from them. The metadata extracted from each XML files includes article DOI, article type, keywords, and timestamps on when the article is received and published. The metadata is then transformed into RDF/XML and enriched with the article PubMed ID and the labels resulting from NER and RE processes (described below) before being added to our knowledge system.

Beyond the metadata part, we also extract raw texts from the XML files by searching for `<p>` tags inside the `<body>` sections, as each `<p>` tag corresponds to a paragraph in the original article. Any texts found inside of `<p>` tags are stripped of formats and hyperlinks. Additionally, image and table captions and math expressions are discarded because they usually do not appear next to the relevant sentences, which can be disruptive to the cohesiveness of the context. We use scispacy's `en_core_sci_sm`[48] to split extracted texts into sentences. The sentences are grouped into paragraphs the same way as they are arranged in the XML files. Local and global span information is calculated for each paragraph, as they are crucial for aligning entities in the downstream tasks.

Besides extracting information from the XML files, we also searched for genetic sequences in the supplemental files to have further insight into each article. Most of sequences are in unstructured format in PDF files, while some are in well-structured format designed to store sequence data such as GenBank, Fasta, and SBOL. The software package SBOL Validator[49] is used to retrieve and identify sequences from these well-structured files. To extract sequences from PDF files, we used `pdftotext`[50] to first convert PDF files into plain texts, and then regular expression to look for plausible sequences. We filtered out sequences that have fewer than six letters to avoid falsely specifying English words as genetic sequences. It is worth mentioning that since some sequences are presented as codons separated by spaces, these spaces are removed, and codons are put together before the filtering.

The extracted metadata, texts, and sequences are organized into one dictionary data

structure per article and saved as JSON files, which are used as input to the subsequent components in our pipeline.

Named Entity Recognition

We define the NER method shown here as a sequence labelling task. For inference, the input is a sequence of words (or tokens) and the output is a sequence of labels and probabilities corresponding to the entity type of each token. Possible labels include B=beginning, I=inside, and O=outside (or not a named-entity of interest). The NER predictions from the fine-tuned BioBERT model which separates the words into subwords. The subwords and their labels are then collapsed into corresponding words and their corresponding predicted labels.

We apply a rule-based label decoder to assign the final labels to tokens and identify entity boundaries. Labels are joined in a greedy manner, i.e. adjacent tokens labeled as the same entity type are joined to make a single entity. After joining adjacent labels, the BIO labels are adjusted such that the first label in a multi-token entity is given a 'B' label, and subsequent labels in the entity are given the 'I' label. This step defines the boundaries of the entity.

Word-level prediction score is assigned the score of its first subword. Each entity is assigned a prediction score that is the average of the scores of its component words. Since separate models are used to predict the different entity types, overlapping predictions arise, for example, when a gene/protein name contains the name of the chemical on which it acts. Overlapping predictions are resolved by choosing the entity with the highest score. We explored different scoring metrics for summarizing the predictions scores at the entity level, including mean, median, and trimmed mean. Using the mean prediction score resolved the problem of overlapping spans.

NER models were fine-tuned and evaluated on each dataset separately. Each model consists of a BERT encoder layer and a linear classification layer with output dimension

equal to the number of labels, which in most cases is three (one for each BIO label). During fine-tuning, the inputs to a model are sequences of subword tokens from the training set, and the outputs are scores from the linear classification layer, one score for each possible label. We use cross entropy loss when propogatin the error back through the network.

We attempt to maximize the number of subword tokens in each input sequences given the constraint of BERT's maximum input size of 512. During pre-processing, we use a sliding window approach to segment a document into non-overlapping sequences of 512 subword tokens (including BERT's special tokens flanking in the input). If the last token of a sequence is in the middle of a known entity, we remove the entity from the current sequence, and place it at the start of the next sequence.

Relation Extraction

The goal of relation extraction (RE) is to determine if there is a relation between terms identified by the NER model, and if there is one, classify the relation. In biomedical RE, we are given a pair of named entities inside the same sentence, and the objective is to determine the biochemical relation between the two entities based on the surrounding context.

For the relation extraction task, we also use BioBERT as this model has been shown to perform very well at processing sequential information. To adapt BioBERT to our specific task of relation extraction, we take the sentence-level context (i.e., the embedding of the <CLS> token, H_{CLS}) generated by BioBERT and feed it into a neural network classifier to determine the type of relation between the named entities in the sentence (Figure 4). This top classifier model consists of three fully connected layers with Rectified Linear Unit (ReLU) activation. A dropout layer with dropout probability of 0.1 is inserted before the first layer. The dimensions of the layers are 768×512, 512×512, and 512×4. We train the classifier to recognize three types of relations (UpRegulator, DownRegulator, and Subtrate) as well as NoRelation, as described above.

To avoid unstable update of BioBERT's parameters in the early stage of training, we keep

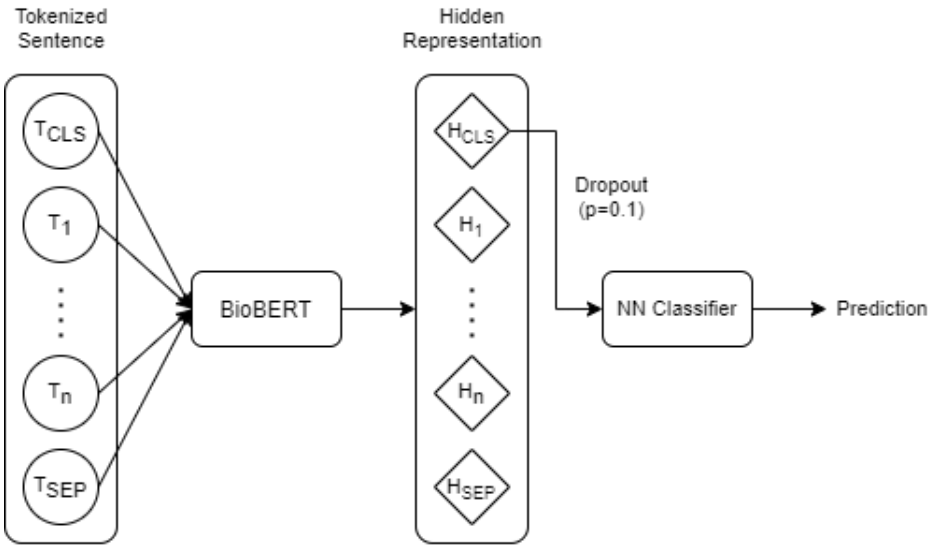

Figure 4: RE model structure

the parameters of BioBERT frozen and only update the top model. Once the top model is trained, we unfreeze BioBERT's parameters and fine-tune the entire model end-to-end. In both training stages, we select the model with the best performance on the validation data to prevent overfitting.

Concept Grounding

At the time of writing, this aspect of the text mining pipeline is just beginning its earliest stages of prototyping. During the initial stage, many of the entities and relationships will be manually curated for a subset of the data in order to create a training dataset which can be used with a series of different classification algorithms.

The first of these classifiers is intended to cluster entity and relationship labels into pools of candidate synonyms. These clusters of labels can then be reconciled against existing synonymous labels within target ontologies and taxonomies relevant to the concepts they represent.

An automated process that leverages existing RESTful Web services and APIs for the ontologies and taxonomies will eventually be developed and implemented as part of the

pipeline to pull in the URL identifying each entity and relationship concept in the context of its ontology or taxonomy. This process will also retrieve the preferred text labels for these concepts. Both URLs and labels will be integrated into RDF/XML metadata files currently being produced for the knowledge system.

Once the first stages have been successfully completed, the second stage of development is to build a second classifier that will be able to automate the step of selecting the appropriate relevant ontology or taxonomy to examine for each entity or relationship, making it possible to automate much of the grounding process.

Topic Modeling

We use Latent Dirichlet Allocation (LDA)[42] for Topic mModeling. The LDA model can be represented by a graphical probabilistic model with three levels.The inner level represents the word level: w denotes a specific word in a particular document, while z denotes the specific topic sampled for that particular word. At the document level, θ represents the topic distribution for a particular document. At the outer corpus level, α and β represents the document topic density and the word topic density, respectively. LDA uses a generative probabilistic approach to model each topic as a mixture of a set of words and each document as a mixture of a set of topics to determine the different topics that a corpus represents and how much of each topic is present in each document.

The specific implementation of LDA we use is the LDA Mallet model,[51] available in the Gensim package.[52] In our LDA model, the number of topics is set to five, which we found experimentally to be stable and easy to interpret.

Preprocessing is often needed to standardize the input to the topic model. We use the BERT BasicTokenizer from Huggingface,[53] which tokenizes the article text by splitting sentences on white spaces, punctuations, and control characters. We also added steps to convert the text to lower case, remove accents, and lemmatize each token using modules from Natural Language Kit (NLTK).[54] Generic terms such as *cell* and *system* commonly

found in the majority of ACS articles obfuscate the process of separating documents and topics. Thus, to remove these generic terms, the top 0.5 percent of terms, based on the TF-IDF score[55] are removed as part of preprocessing to refine the specificity of the topic model.

To tailor LDA for synthetic biology text, our topic modeling process treats NER terms as special terms that are not preprocessed. NER terms are named entities identified by the NER model as being of an entity type of interest, as described above. NER terms consisting of multiple words are connected by underscore to keep the entire term intact. For some NER terms, substrings are also identified as NER terms. For example, both *aspartic acid* and *tetra-aspartic acid* are NER terms. Our preprocessing checks for these cases, and prioritizes the longest NER term to be kept intact. Keeping NER terms intact allows for these named entities to influence the topics that are discovered by the topic models.

Results and Discussion

Named Entity Recognition

We evaluate our NER system using precision, recall and F_1 score. Precision is the ratio between correctly predicted mentions over the total set of predicted mentions for a specific entity; recall is the ratio of correctly predicted mentions over the actual number of mentions; and F_1 is the harmonic mean between precision and recall. We report both the strict and lenient results for each entity category. In strict evaluation, two annotations are equal only if they have the same tag with exactly matching spans. With the lenient evaluation, two annotations are equal if they share the same tag and their spans overlap by at least one character; when two or more entities in the system overlap with one entity in the ground truth (or vice versa), all sharing the same tag, only one pair is counted and the extra entity is ignored entirely. The NER model hyperparameters (e.g. epoch, dropout) were tuned over the HUNER development set. The parameters that obtained the best performance on the

development set were used to obtain the results for the HUNER test data and adjudicated ACS data set.

Table 8 shows the exact and lenient precision (P), recall (R) and F1 scores for our NER system when trained over the HUNER training data set and evaluated over the HUNER test data set and our adjudicated ACS data set. The results are entity dependent. For Cell Line, the current datasets are not able to generalize to the extraction of cell Line for synthetic biology. Cell Line mentions in the ACS data set include *E. coli DH1*, *E. coli JAD*, *MG1655*, and *JAD-1*. In contrast, Cell Line mentions in the training data include *rat macrophage cell line R2*, *CD4 negative T cell lines*, and *Jurkat cells*. Species mentions in the ACS dataset contain mentions including *Pseudomonas butanovora*, *Zymonomas mobilis*, and *Lactococcus lactis*, while the Species training data contains mentions including *human* and *mouse* in the Variome dataset, and *Xenopus laevis*, *Caenorhabditis elegans*, and *bacteriophage M13* in the Linneaus dataset. Not surprisingly, the model trained on Variome species struggles to identify entities written in binomial nomenclature, which is the most common case in the ACS dataset. This indicates that the creation of domain-specific training data is necessary for synthetic biology and that models trained on biomedical entities such as Cell Line and Species do not generalize well to other biological domains. However, models trained on Chemicals and Genes tend to identify entities well, and the difference in exact and lenient scores indicate a need for better post-processing.

Relation Extraction

To train our model, we use the ChemProt corpus[47] to train and evaluate our relation extraction model. Each sample inside the ChemProt corpus consists of a sentence, where a Chemical entity and a Gene entity are replaced with generic tokens ("@CHEMICAL$" and "@GENE$" respectively), and a label, which indicates the ground truth relation between the Chemical and the Gene entities. To perform relation extraction on the ACS texts, we split the texts into sentences (which is performed by the Text Parsing component described

Table 8: The exact and lenient precision (P), recall (R) and F1 score for NER trained on the HUNER training set and evaluated over the HUNER test set and the adjudicated ACS test set. Overall average over each of the entity types are included.

			HUNER						ACS					
			EXACT			LENIENT			EXACT			LENIENT		
Entity	**Dataset**	Epoch	P	R	F1	P	R	F1	P	R	F1	P	R	F1
CELL LINE	CLL	19	0.7922	0.8472	0.8188	0.8571	0.9167	0.8859	0.2727	0.5000	0.3529	0.5000	0.9167	0.6471
	GELLUS	10	0.6437	0.8503	0.7327	0.7409	0.9839	0.8453	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	JNLPBA	9	0.7010	0.6675	0.6838	0.8317	0.7954	0.8131	0.3182	0.3043	0.3111	0.3182	0.3043	0.3111
	average		0.7123	0.7883	0.7451	0.8099	0.8987	0.8481	0.1970	0.2681	0.2213	0.2727	0.4070	0.3194
CHEMICAL	CDR	11	0.8904	0.8803	0.8853	0.9430	0.9350	0.9390	0.7654	0.7248	0.7445	0.8586	0.8131	0.8352
	CEMP	1	0.8411	0.8219	0.8314	0.9167	0.8886	0.9024	0.7325	0.7868	0.7587	0.8059	0.8627	0.8333
	CHEBI	15	0.7442	0.7555	0.7498	0.9098	0.9095	0.9097	0.5998	0.7160	0.6527	0.7072	0.8409	0.7683
	CHEMDNER	6	0.8768	0.9007	0.8886	0.9408	0.9525	0.9466	0.6886	0.7763	0.7298	0.7873	0.8853	0.8334
	average		0.8381	0.8396	0.8388	0.9276	0.9214	0.9244	0.6966	0.7510	0.7214	0.7898	0.8505	0.8176
GENE	BC2GM	15	0.7857	0.7698	0.7777	0.9692	0.9315	0.9500	0.4450	0.5425	0.4889	0.6992	0.7912	0.7424
	BIOINFER	19	0.7744	0.8065	0.7901	0.9465	0.9562	0.9513	0.4620	0.3025	0.3656	0.7302	0.4591	0.5637
	DECA	16	0.7703	0.6357	0.6965	0.8553	0.7088	0.7752	0.3101	0.4107	0.3534	0.4729	0.5980	0.5281
	FSU	19	0.8924	0.8701	0.8811	0.9756	0.9373	0.9561	0.5256	0.7451	0.6164	0.5705	0.8160	0.6715
	GPRO	6	0.7556	0.6429	0.6947	0.8739	0.7444	0.8040	0.1457	0.6861	0.2404	0.2078	0.9306	0.3397
	IEPA	13	0.8900	0.8344	0.8613	0.9567	0.8885	0.9213	0.1426	0.4126	0.2120	0.1798	0.5179	0.2670
	JNLPBA	7	0.7990	0.7718	0.7852	0.9271	0.8809	0.9034	0.3194	0.5037	0.3909	0.5194	0.7267	0.6058
	MIRNA	13	0.7938	0.5908	0.6774	0.9485	0.7041	0.8082	0.0667	0.5733	0.1194	0.0884	0.7703	0.1586
	OSIRIS	18	0.7595	0.7595	0.7595	0.8660	0.8720	0.8690	0.4403	0.5954	0.5062	0.5395	0.7160	0.6154
	VARIOME	19	0.9214	0.8792	0.8998	0.9806	0.9283	0.9537	0.0248	0.6957	0.0479	0.0357	0.9200	0.0687
	average		0.8142	0.7561	0.7823	0.9299	0.8552	0.8892	0.2882	0.5468	0.3341	0.4043	0.7246	0.4561
SPECIES	LINNEAUS	16	0.7626	0.8000	0.7808	0.8417	0.8731	0.8571	0.9596	0.8425	0.8973	1.0000	0.8780	0.9350
	MIRNA	11	0.5903	0.6943	0.6381	0.6520	0.7668	0.7048	0.0090	0.0238	0.0130	0.0179	0.0476	0.0261
	S800	15	0.6955	0.6809	0.6882	0.8538	0.8382	0.8459	0.9058	0.8745	0.8899	0.9910	0.9567	0.9736
	VARIOME	19	0.1515	0.4348	0.2247	0.1515	0.4348	0.2247	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	average		0.5500	0.6525	0.5830	0.6248	0.7282	0.6581	0.4686	0.4352	0.4501	0.5022	0.4706	0.4837
OVERALL AVERAGE			0.7539	0.7569	0.7498	0.8542	0.8498	0.8460	0.3873	0.5246	0.4139	0.4776	0.6548	0.5107

above), and use the results from the NER task to identify Chemical and Gene entities in each sentence. For each identified Chemical-Gene pair, we generate an input sentence with the pair replaced by generic tokens. For example, if a sentence contains three chemicals and two genes, we generate six input sentences for all combinations of Chemical-Gene pairs.

Table 9: Relation Extraction results for ChemProt test data & ACS data

	ChemProt			ACS		
	Precision	Recall	F1	Precision	Recall	F1
UpRegulator	0.7160	0.6977	0.7068	0.8571	0.2500	0.3871
DownRegulator	0.7594	0.8037	0.7809	0.7813	0.6410	0.7042
Subtrate	0.6077	0.6351	0.6211	0.8381	0.5906	0.6929
Overall Average	0.7171	0.7434	0.7300	0.8264	0.5613	0.6685

Table 9 shows the overall and class-specific precision, recall, and F_1 scores of the relation extraction model over the ChemProt test set and the ACS dataset when trained on the ChemProt training dataset. Here, precision is the ratio between correctly predicted relations to the total set of predicted relations; recall is the ratio of correctly predicted relations to the actual number of relations, and F_1 is the harmonic mean between the two. The results show that the relation *DownRegulator* obtains a higher F_1 score than the other two relations. The results also show that the RE model is able to generalize to the ACS data with high precision, though with lower recall, decreasing the overall F_1 score.

Concept Grounding

A preliminary manual mapping was created for the NER results generated for one exemplar article from the ACS dataset. NER terms generated for Species and Cell Line type mentions were reconciled against the NCBI's Species taxonomy. Of the 24 unique mentions, 10 (41.67%) were aligned with classes existing in the taxonomy. Of the remaining mentions, 11 (45.83%) were not found in the taxonomy and 4 (16.67%) were too ambiguous to differentiate from multiple existing entries in the taxonomy. In an attempt to improve the grounding results, the Cel Lline mentions were grouped with the Gene mentions and reconciled against

MeSH terms. Of this group of 100 unique mentions, 49 (49.00%) were successfully aligned with existing MeSH concepts. Of the remaining 51 mentions, 40 (40.00%) were not found, 4 (4.00%) were too ambiguous to differentiate among multiple different matching MeSH terms, and 7 (7.00%) were aligned with MeSH terms that were linked with Species terms not mentioned in the article, and were not considered to have been successfully grounded to useful concepts. Finally, NER mentions that were classified as chemical mentions were reconciled against the CHEBI ontology. Of these 29 unique mentions, 22 (75.86%) were successfully aligned with matching concepts in the CHEBI ontology. Of the remaining 7 mentions, 5 (17.24%) could not be aligned with concepts in the ontology and 2 (6.90%) were too ambiguous to differentiate among multiple matching concepts in the ontology.

The manual concept grounding failed in many instances for a variety of reasons. However, the value additions created by linking NER mentions to corresponding concepts in ontologies, subject thesauri, and taxonomies seem achievable for a sizable proportion of the NER dataset. Next steps for this work will primarily comprise of building a workflow to simplify the grounding work and allow to manageably be applied across the entire NER pipeline output.

Topic Modeling

Since LDA is unsupervised, as is the case in general with topic modeling techniques, there is no ground truth to specify the correct topics of a corpus. Thus, to evaluate the topic model's results, a domain expert on our team provided interpretations of the abstract topics uncovered by our LDA model by examining the top high-scoring words associated with each abstract topic. The expert also provided a topic relatedness score which measures how well the top words uncovered for each topic relates to the topic interpretation. We also calculated the coherence score[56] and used that as an additional metric to evaluate the resulting topics. The coherence score measures the degree of semantic similarity among the high contribution terms in the topic, and has a value between zero and one. We compared results with the article abstracts included or excluded as shown in Table 10. We found that

including the abstracts resulted in a higher topic relatedness score from our expert but a lower coherence score. Both relatedness and coherence scores are close, however, which indicates that including abstracts does not have a significant impact on topic modeling results.

Table 10: Topic modeling results. Topic Interpretation and Relatedness scores are provided by a synthetic biology expert. Results with and without using abstracts are included.

Topic Interpretation	LDA (k=5 without abstract)	Relatedness (1=not related, 5=very related)	LDA (k=5 with abstract)	Relatedness (1=not related, 5=very related)
genetic circuit design	circuit, input,network, output, behavior, gate, strand, simulation, device...	4.0	circuit, network, output, input, behavior, gate, strand, device, ...	4.0
gene expression	terminator, mrna, bp, GFP, repression, tRNA, cassette, ...	3.5	mrna, terminator, repression, GFP, strength, ribosome, ...	3.5
metabolic engineering	biosynthesis, titer, carbon, cluster, metabolite, fermentation, flux, deletion...	5.0	biosynthesis, titer, metabolite, carbon, fermentation, flux, ...	5.0
protein-related/ strain construction	ion, membrane, residue, peptide, surface, affinity, ...	3.0	yeast, cassette, tRNA, residue, bp, CRISPR, recombination, cluster, DNA, ...	4.0
biosensors	sensor, light, activation, ligand, GFP, switch, biosensor, riboswitch, receptor, ...	4.5	light, domain, activation, ligand, ion, fusion, switch, sensor, ...	4.5
Average Relatedness Score		4.0		4.2
Coherence Score		0.453		0.424

Use Cases

Table 11 shows four use case type questions the SBKS Text Mining Pipline would be able to answer. The table contains the *Use Case* question potentially asked by the user. The

Input describes that type of information required by the system. The *Ouput* describes the type of information returned to answer the user's question. The *Linked Data* describes the meta data information that can be returned to the user for further exploration associated with the question. The *Pipeline Components* identifies what components of the pipeline are used to answer the question.

Table 11: SBKS Text mining pipeline use cases

Use Case:	Expressing a gene in a known organism	Answering "has this been done before?" questions	Extracting parts	Exploring native organism biology
Input	Host, species or strain	Part, species, or strain	Species or strain with part (promoter or gene)	Species or strain
Output	Plasmids, promoters, other expression parts that work in that organism (e.g. qPCR data)	Characterization data (e.g. HPLC titers, flow cytometery data), plasmicds	Sequences	Metabolic models, moics data (e.g. RNAseq, meabolome, proteome, genome)
Linked Data	Names of parts or sequences, physical repositories (e.g. Addgene, ATCC), transformationi protocols (papers), author/institute names, conference venues	Papers, repositories (e.g. Addgene, AtCC), author names, conference venues	Names of parts or sequences, repositories	NCBI genome, NCBI Geo datasets, GSMMS, RNAseq DGE tables
Pipeline Components	NER & RE & Grounding	NER & RE & Grounding	NER & RE & Grounding	NER & Topic Modeling

Conclusion and Future Work

In this paper, we have described our preliminary SBKS text mining pipeline to extract and correlate synthetic biology information from the literature. In the future, we plan to expand each of our components.

State-of-the-art methods for NER utilize transformers to encode text for downstream entity classification. One drawback is the size of the models and the computational resources required to fine-tune and perform inference on text. As the number of desired entity classes grows, the task of performing inference becomes more time and resource consuming. Recent work in multi-task learning helps mitigates this issue by sharing the transformer encoder layer between all NER tasks,[5] which makes the problem scalable. Another possible by-product of this approach is shared information learned in the encoder layer that may improve the performance on a task compared to a single task model.[57] We have implemented a multi-task model and are testing its performance on inference tasks described in this paper. Future work with multi-task task learning also includes jointly learning NER and RE, as well as extending BioBERT's pre-training data to include domain-specific synthetic biology articles and abstracts. We also plan to utilize our current grounding data to aid in automating the process of grounding entities to their respective ontologies.

For topic modeling, our topics are currently used to tag articles. However, topic modeling can also be used to analyze trends over time, to study how terms and/or topics in synthetic biology articles evolve over the years. Additionally, we are working on ways to incorporate topic modeling results into an intuitive user interface to browse and explore SKBS contents.

Funding

This work was funded by the National Science Foundation under Grants No. 1939885, 1939951, 1939860, and 1939929. The computer resources used for this work were provided in part by NSF award numbers 1730158, 2100237, and 2120019.

Acknowledgement

COnflict of Interests: None

The authors would like to thank Jeff Elhai, Chris Meyers, Brandon Sepulvado, and Xiang Lu for their collaboration and valuable insights in developing the methodologies presented in this paper.

BTM: contributed to the overall writing of the manuscript, data set development and analysis, and the development of the NER and RE modules. MHN: contributed to the overall writing of the manuscript, data set development and analysis, and the development of the RE and Topic Modeling modules. NER: NER writing, model development and data set development. JT: RE writing, model development and data set development. SR: RE writing, model development and data set development. DX: Topic modeling writing, and model development. YH: Topic modeling writing, and model development. GN: NER model development. JSD: concept grounding writing and model development. JJ: concept grounding writing and model development. KK: data set development and analysis. EY: data set development and analysis.

References

(1) Mante, J. et al. The synthetic biology knowledge system. *ACS Synthetic Biology* **2021**,

(2) Chen, L.; Liu, H.; Friedman, C. Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics* **2005**, *21*, 248–256.

(3) Rodriguez-Esteban, R. Semantic persistence of ambiguous biomedical names in the citation network. *Bioinformatics* **2020**, *36*, 2224–2228.

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

(5) Akdemir, A.; Shibuya, T. Analyzing the Effect of Multi-task Learning for Biomedical Named Entity Recognition. *arXiv:2011.00425 [cs]* **2020**, arXiv: 2011.00425.

(6) Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C. H.; Kang, J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2020**, *36*, 1234–1240.

(7) Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation* **1997**, *9*, 1735–1780.

(8) Lafferty, J.; McCallum, A.; Pereira, F. C. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. **2001**,

(9) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. Advances in neural information processing systems. 2017; pp 5998–6008.

(10) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]* **2019**, arXiv: 1810.04805.

(11) Li, J.; Sun, A.; Han, J.; Li, C. A Survey on Deep Learning for Named Entity Recognition. *arXiv:1812.09449 [cs]* **2020**, arXiv: 1812.09449.

(12) Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C. H.; Kang, J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2019**, btz682.

(13) Morgan, L. Z. W. X. C. A. F. J. R. P. D. A. F. K. L. R. H. J., A.A.; Sun, C. Overview of BioCreative II gene normalization. *Genome Biology* **2008**, *9*, 1–9.

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

(15) LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W.; Jackel, L. D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation* **1989**, *1*, 541–551.

(16) Zhang, Y.; Lin, H.; Yang, Z.; Wang, J.; Sun, Y.; Xu, B.; Zhao, Z. Neural network-based approaches for biomedical relation classification: A review. *Journal of Biomedical Informatics* **2019**, *99*, 103294.

(17) Krallinger, M. et al. Overview of the BioCreative VI chemical-protein interaction Track. *Proceedings of BioCreative III Workshop* **2017**, 141–146.

(18) Henry, S.; Wang, Y.; Shen, F.; Uzuner, O. The 2019 National Natural language processing (NLP) Clinical Challenges (n2c2)/Open Health NLP (OHNLP) shared task on clinical concept normalization for clinical records. *Journal of the American Medical Informatics Association: JAMIA* **2020**, *27*, 1529–1537.

(19) Liu, F.; Shareghi, E.; Meng, Z.; Basaldella, M.; Collier, N. Self-Alignment Pretraining for Biomedical Entity Representations. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2021; pp 4228–4238.

(20) Federhen, S. The NCBI taxonomy database. *Nucleic acids research* **2012**, *40*, D136–D143.

(21) Tutubalina, E.; Kadurin, A.; Miftahutdinov, Z. Fair Evaluation in Concept Normalization: a Large-scale Comparative Analysis for BERT-based Models. Proceedings of the 28th International Conference on Computational Linguistics. 2020; pp 6710–6716.

(22) Aronson, A. R. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Proceedings of the AMIA Symposium. 2001; p 17.

(23) Aronson, A. R. Metamap: Mapping text to the umls metathesaurus. *Bethesda, MD: NLM, NIH, DHHS* **2006**, *1*, 26.

(24) Aronson, A. R.; Lang, F.-M. An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association* **2010**, *17*, 229–236.

(25) Mork, J. G.; Jimeno-Yepes, A.; Aronson, A. R., et al. The NLM Medical Text Indexer System for Indexing Biomedical Literature. *BioASQ@ CLEF* **2013**, *1*.

(26) Limsopatham, N.; Collier, N. Normalising Medical Concepts in Social Media Texts by Learning Semantic Representation. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016; pp 1014–1023.

(27) Tutubalina, E.; Miftahutdinov, Z.; Nikolenko, S.; Malykh, V. Medical concept normalization in social media posts with recurrent neural networks. *Journal of biomedical informatics* **2018**, *84*, 93–102.

(28) Niu, J.; Yang, Y.; Zhang, S.; Sun, Z.; Zhang, W. Multi-task character-level attentional networks for medical concept normalization. *Neural Processing Letters* **2019**, *49*, 1239–1256.

(29) Wajsbürt, P.; Sarfati, A.; Tannier, X. Medical concept normalization in French using multilingual terminologies and contextual embeddings. *Journal of Biomedical Informatics* **2021**, *114*, 103684.

(30) Miftahutdinov, Z.; Tutubalina, E. Deep Neural Models for Medical Concept Normalization in User-Generated Texts. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop. 2019; pp 393–399.

(31) Leaman, R.; Islamaj Doğan, R.; Lu, Z. DNorm: disease name normalization with pairwise learning to rank. *Bioinformatics* **2013**, *29*, 2909–2917.

(32) Ji, Z.; Wei, Q.; Xu, H. BERT-based ranking for biomedical entity normalization. *AMIA Summits on Translational Science Proceedings* **2020**, *2020*, 269.

(33) Liu, H.; Xu, Y. A deep learning way for disease name representation and normalization. National CCF conference on natural language processing and Chinese computing. 2017; pp 151–157.

(34) Li, H.; Chen, Q.; Tang, B.; Wang, X.; Xu, H.; Wang, B.; Huang, D. CNN-based ranking for biomedical entity normalization. *BMC bioinformatics* **2017**, *18*, 79–86.

(35) Broscheit, S. Investigating Entity Knowledge in BERT with Simple Neural End-To-End Entity Linking. Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL). 2019; pp 677–685.

(36) Sung, M.; Jeon, H.; Lee, J.; Kang, J. Biomedical Entity Representations with Synonym Marginalization. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020; pp 3641–3650.

(37) Phan, M. C.; Sun, A.; Tay, Y. Robust representation learning of biomedical names. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019; pp 3275–3285.

(38) Mondal, I.; Purkayastha, S.; Sarkar, S.; Goyal, P.; Pillai, J.; Bhattacharyya, A.; Gattu, M. Medical Entity Linking using Triplet Network. Proceedings of the 2nd Clinical Natural Language Processing Workshop. 2019; pp 95–100.

(39) Fakhraei, S.; Mathew, J.; Ambite, J. L. Nseen: Neural semantic embedding for entity normalization. Joint European Conference on Machine Learning and Knowledge Discovery in Databases. 2019; pp 665–680.

(40) Xu, D.; Zhang, Z.; Bethard, S. A Generate-and-Rank Framework with Semantic Type Regularization for Biomedical Concept Normalization. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020; pp 8452–8464.

(41) Steyvers, M.; Griffiths, T. Probabilistic topic models. *Handbook of latent semantic analysis* **2007**, *427*, 424–440.

(42) Blei, D. M.; Lafferty, J. D. Dynamic Topic Models. Proceedings of the 23rd International Conference on Machine Learning. New York, NY, USA, 2006; p 113–120.

(43) Jelodar, H.; Wang, Y.; Yuan, C.; Feng, X.; Jiang, X.; Li, Y.; Zhao, L. Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications* **2019**, *78*, 15169–15211.

(44) Wang, H.; Ding, Y.; Tang, J.; Dong, X.; He, B.; Qiu, J.; Wild, D. J. Finding Complex Biological Relationships in Recent PubMed Articles Using Bio-LDA. *PLOS ONE* **2011**, *6*, 1–14.

(45) Weber, L.; Münchmeyer, J.; Rocktäschel, T.; Habibi, M.; Leser, U. HUNER: improving biomedical NER with pretraining. *Bioinformatics* **2020**, *36*, 295–302.

(46) Taboureau, O.; Nielsen, S. K.; Audouze, K.; Weinhold, N.; Edsgärd, D.; Roque, F. S.; Kouskoumvekaki, I.; Bora, A.; Curpan, R.; Jensen, T. S., et al. ChemProt: a disease chemical biology database. *Nucleic acids research* **2010**, *39*, D367–D372.

(47) Sun, C.; Yang, Z.; Su, L.; Wang, L.; Zhang, Y.; Lin, H.; Wang, J. Chemical-protein Interaction Extraction via Gaussian Probability Distribution and External Biomedical Knowledge. *Bioinformatics* **2020**, btaa491.

(48) Neumann, M.; King, D.; Beltagy, I.; Ammar, W. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. Proceedings of the 18th BioNLP Workshop and Shared Task. Florence, Italy, 2019; pp 319–327.

(49) A Validator and Converter for the Synthetic Biology Open Language. *6*.

(50) Palmer, J. A. pdftotext: Simple PDF text extraction. `https://github.com/jalan/pdftotext`.

(51) McCallum, A. K. MALLET: A Machine Learning for Language Toolkit. `http://mallet.cs.umass.edu`.

(52) Řehůřek, R. Gensim: Topic Modelling for Humans. `https://radimrehurek.com/gensim/index.html`.

(53) Hugging Face: BertTokenizer. `https://huggingface.co/transformers/model_doc/bert.html#berttokenizer`.

(54) Project, N. Natural Language Toolkit. `https://www.nltk.org/`.

(55) Schütze, H.; Manning, C. D.; Raghavan, P. *Introduction to information retrieval*; Cambridge University Press Cambridge, 2008; Vol. 39.

(56) Röder, M.; Both, A.; Hinneburg, A. Exploring the space of topic coherence measures. Proceedings of the eighth ACM international conference on Web search and data mining. 2015; pp 399–408.

(57) Peng, Y.; Chen, Q.; Lu, Z. An Empirical Study of Multi-Task Learning on BERT for Biomedical Text Mining. *arXiv:2005.02799 [cs]* **2020**, arXiv: 2005.02799.

Graphical TOC Entry