Database, 2022, 1–6
DOI: https://doi.org/10.1093/database/baac066
Original article

Pre-trained models, data augmentation, and ensemble learning for biomedical information extraction and document classification

Arslan Erdengasileng[1], Qing Han[1], Tingting Zhao[2], Shubo Tian[1], Xin Sui[1], Keqiao Li[1], Wanjing Wang[1], Jian Wang[3], Ting Hu[1], Feng Pan[1], Yuan Zhang[1] and Jinfeng Zhang[1,*]

[1]Department of Statistics, Florida State University, Tallahassee, FL 32306, USA
[2]Department of Geography, Florida State University, Tallahassee, FL 32306, USA
[3]Cloudmedx Inc, Palo Alto, CA 94301, USA

[*]Corresponding author: Tel: +850 644 3218; Fax: +850 644 5271; Email: jinfeng@stat.fsu.edu

Citation details: Erdengasileng, A., Han, Q., Zhao, T. *et al.* Pre-trained models, data augmentation, and ensemble learning for biomedical information extraction and document classification. *Database* (2022) Vol. 2022: article ID baac066; DOI: https://doi.org/10.1093/database/baac066

Abstract

Large volumes of publications are being produced in biomedical sciences nowadays with ever-increasing speed. To deal with the large amount of unstructured text data, effective natural language processing (NLP) methods need to be developed for various tasks such as document classification and information extraction. BioCreative Challenge was established to evaluate the effectiveness of information extraction methods in biomedical domain and facilitate their development as a community-wide effort. In this paper, we summarize our work and what we have learned from the latest round, BioCreative Challenge VII, where we participated in all five tracks. Overall, we found three key components for achieving high performance across a variety of NLP tasks: (1) pre-trained NLP models; (2) data augmentation strategies and (3) ensemble modelling. These three strategies need to be tailored towards the specific tasks at hands to achieve high-performing baseline models, which are usually good enough for practical applications. When further combined with task-specific methods, additional improvements (usually rather small) can be achieved, which might be critical for winning competitions.

Database URL: https://doi.org/10.1093/database/baac066

Introduction

The scientific publications in biomedical sciences have been increasing in volume with an accelerated speed. The amount of the scientific literature has posed a daunting challenge for researchers to find relevant information for their research. Failing to identify the most relevant information may cause misinterpretation of experimental results or wasted time and/or resources on duplicated works.

To deal with this challenge, automatic information extraction and document classification methods have been developed over the years with various successes (1–13). To evaluate these methods and facilitate the continued development, BioCreative Challenge was established in 2007 to test the information extraction methods in biomedical domain (14–17).

In the latest challenge, BioCreative Challenge VII, there are five different tracks:

Track 1: DrugProt: Text-mining drug/chemical–protein interactions.
Track 2: National Library of Medicine (NLM)-Chem Track: Full-text Chemical Identification and Indexing in PubMed articles.
Track 3: Automatic extraction of medication names in tweets.
Track 4: Coronavirus Disease 2019 (COVID-19) text-mining tool interactive demo.
Track 5: LitCovid track multi-label topic classification for COVID-19 literature annotation.

Our team, FSU2021, participated in all the five tracks. Overall, our methods performed well for all the tracks, especially for Tracks 2 and 3. In Track 2, we ranked the second for the first two sub-tasks and ranked the best for the third sub-task. In Track 3, we ranked the fourth. Except for Track 4 which did not score the participating tools, our methods performed substantially better than the median scores and baselines.

In the past couple of years, the pre-trained language models have become the mainstream for many natural language processing (NLP) tasks and achieved the state-of-the-art performances. They formed the baselines for all our NLP models. On top of the baseline models, we explored two general strategies: data augmentation and ensemble models. Data augmentation was used for all the tracks except Track 4 and it is effective for Tracks 1, 2 and 3. Ensemble models were used

Received 31 March 2022; Revised 29 July 2022; Accepted 9 August 2022

for Tracks 1 and 5 and it helped in both cases. In addition, we tried some track-specific strategies, especially for Track 2, which are described in detail in Methods.

In this paper, we describe our works in these five tracks and discuss the strategies that worked and the lessons we have learned. The experiences and lessons learned in the BioCreative Challenge VII helped us to achieve the best overall score on the LitCoin NLP Challenge held from November 2021 to February 2022 organized by the National Center for Advancing Translational Sciences (NCATS) and the National Aeronautics and Space Administration (NASA) (18).

Data and methods

The data used in BioCreative Challenge VII have been described previously in the proceeding papers of the workshop and also by overview papers in the Database special issue (19–26).

We organize the methods used by the types of the methods, instead of tracks. For the details on the methods used in each track, please refer to the workshop proceeding papers.

Selection of pre-trained models

We tried different per-trained models on different tracks, including Bidirectional Encoder Representations from Transformers (BERT) (27) (Tracks 1, 2, 3 and 5), BioBERT (28) (Track 1, 2, and 3), PubMedBERT (29) abstract only (Track 5), PubMedBERT fulltext [Tracks 1, 2, 3 (abstract only and full text) and 5], Sentence BERT (30) (Track 1), T5 (31) (Track 1), BlueBERT (32) (Track 2), SciBERT (33) (Track 2), RoBERTa (34) and ClinicalBERT (35) (Track 2).

Data augmentation

Several data augmentation strategies were used as follows:

1. For Track 1, after training the BERT model, we obtained a subset of training data that the trained model predicted wrong. For every sentence, we used StanfordNLP to get the shortest path between the chemical name and gene name and then randomly deleted one word not in the path. We labelled them the same as the originals and augmented training data sets with these sentences in all our models.
2. For Track 2, we first replaced the chemical entities with random strings (i.e. Aspirin→badjaxfjfg). We also randomly selected one nonchemical entity in sentences which contain chemical entities and then replaced it with a random string (i.e. that →hsw).
3. For Track 3, we considered three different data augmentation strategies:
   a. Augment true cases by replacing each original true entity with a randomly chosen medication mention from the pool. The medication mention pool could be generated from either BioCreative_TrainTask3.0 and 3.1 two data sets or BioCreative_TrainTask3.0 and 3.1 and SMM4H18 three data sets.
   b. Augment true cases by replacing each original true entity with a random string. The string contains 3–10 characters randomly selected from a–z to A–Z.
   c. Augment true cases by dropping a randomly selected word which is not or not part of a true entity.
4. For Track 5, we tried the following strategies:
   d. Double cases for all instances by removing 10 words with lowest term frequency–inverse document frequency (TF-IDF);
   e. Double cases for all instances by removing 10 words with highest TF-IDF;

Each of the strategy has its own advantages in terms of adding additional useful information for the model training. For example, replacing an entity with another entity of the same type is relatively safe as it uses only the true entities, but it may not ‘squeeze out’ enough information as using the random strings. Replacing an entity with random strings is more aggressive by forcing the model to learn more about the context information by introducing new names the pre-trained models have never seen before. But it may introduce some unwanted bias. Dropping a non-essential word introduces more diversity into the context itself although it may bring in more grammar mistakes or missing words that are still important for classification. For each of the strategies, there are options of augmenting all or partial data; and the number of rounds the data would be augmented based on positive cases. A combination of different strategies could also bring compound effects into the model training.

Ensemble models

We tried ensemble models for Tracks 1 and 5. In Track 1, we also experimented with training additional machine learning models using outputs of individual models.

The first and simplest idea is using the majority vote from the output of individual models. When all the models predict differently, we use the result from the one that yields the highest F1 score in the training data.

In Track 1, we also experimented with the idea of training ensemble models based on the clustering results from the Sentence BERT model and Hierarchical Density-based Spatial Clustering of Applications with Noise (HDBSCAN). If the performances of the models vary from cluster to cluster, such ensemble methods can learn this pattern and adjust the weights of the models dynamically according to the cluster of any given sentence. Inspired by this idea, we implemented two ensemble models: (i) simple ensemble and (ii) trained ensemble. For the simple ensemble method, we assigned each cluster a model, which yielded the best accuracy over all the sentences in the cluster for the training data. We found using the accuracy metric instead of the F1 score to select models yielded the better result in our experiments. When making predictions, we first predicted the cluster of the sentence and then used the model assigned to that cluster to make the final prediction. For the trained ensemble method, we extracted the following features from our models (because the T5 model was added in the last week of the competition, we did not have enough time to incorporate its results into the ensemble method): (i) last layer features of the BERT model; (ii) class probabilities of the BERT model; (iii) cluster ID from the HDBSCAN and (iv) class label from *k*-NN classification. We trained an ensemble model of (i) XGBoost (11); (ii) logistic regression; (iii) Extra Trees classifier and (iv) Random Forest classifier, to predict one among the following four scenarios: (i) both BERT and Sentence BERT predicted wrong; (ii) only BERT predicted the label right; (iii) only Sentence BERT predicted the label right and (iv) both models predicted the label right. During inference, for sentences that fall into the first scenario, we used the predicted label from the model that yielded the better F1 score overall.

For Track 5, we trained multiple models and took the topic label probabilities predicted by different models as input and computed an average of the predicted probabilities of each topic label as the final prediction.

Task-specific methods

In addition to the above strategies, we also tried some task-specific methods. In this paper, we will focus on Track 2, for which we have performed very well overall (36).

Track 2 has three sub-tasks: (i) named entity recognition; (ii) chemical normalization and (iii) chemical indexing. We ranked the second on the first two sub-tasks and ranked the first on the third sub-task.

For the named entity recognition (NER) sub-task, we first used Ab3P (37), an abbreviation definition detector trained on PubMed abstracts, to recognize abbreviations in the text. The full names and their abbreviations are linked within the same articles, and all the occurrences received the same NER label. We checked the consistency for the same term in an article, since they should be annotated as the same type, except overlapping with other entities.

We also trained another BioBERT-based protein NER model to detect protein entities. The goal was to further remove wrongly labelled chemical names which are part of protein names. The reason for using BioBERT is because this model performed better for protein NER task. The rule is:

1. If a token word was recognized by PubMedBERT-based Chemical NER model as a chemical entity and by BioBERT-based protein NER model as a protein entity at the same time, its predicted entity label ('B' or 'I') would be changed 'O'.
2. If a predicted chemical entity name was followed by a predicted protein entity name, then the predicted chemical entity label ('B' or 'I') would be changed to 'O'.

For chemical normalization, we built a sieve-based pipeline using multiple dictionaries including supplementary concept records, MRCONSO, PubTator (38) and NLM-Chem.

For chemical indexing, we built a model to predict the indexing status of individual Medical Subject Headings (MeSH) IDs by extracting features from the articles, where the MeSH IDs were identified. We dealt with one MeSH term at a time by predicting whether it should be used for indexing or not. To remove the noise from the long text, we broke up full texts into sentences and selected the sentences with chemical mentions of the corresponding MeSH terms as input to the model. The labels are simply True or False based on if the MeSH terms were used for indexing the articles or not. We added engineered features before the sentences, such as the section where the sentences were taken from and the chemical names whose MeSH terms were to be predicted.

Table 1. Results using different pre-trained models. The values in the table are F1 scores on test data

Pre-trained models	Track 1	Track 2	Track 3	Track 5
BERT	–	0.8018	–	–
BioBERT	0.683	0.8433	0.5957	0.9067
PubMedBERT abstract	–	–	0.5922	0.9027
PubMedBERT fulltext	0.732	0.8679	0.6257	0.9066
BlueBERT	–	0.8442	–	0.8956
SciBERT	–	0.8495	–	–
ClinicalBERT	–	0.8114	–	–
T5	0.739	–	–	–
RoBERTa		0.8536		

Table 2. Performance of different data augmentation strategies for four tracks. Not all the data augmentation methods were tried for all the tracks due to the differences in the data/tasks. * F1 score on validation data

Data augmentation methods	Track 1	Track 2	Track 3	Track 5
No data augmentation	0.721	0.8711	0.7090	0.9298*
Dropping a non-essential word	0.749		0.7913	–
Replacing words with random strings	–	0.8744	0.800	–
Replacing an entity name with another name of the same type	–	–	0.837	–
Dropping words with lowest TF-IDF values	–	–	–	0.9271*
Dropping words with highest TF-IDF values	–	–	–	0.9257*

Results

We organize the results part first by the methods we used. We then present some detailed results for Tracks 2 and 3, for which our team performed well.

Pre-trained models

Table 1 shows the results using different pre-trained models for the four tracks. Clearly PubMedBERT works better than other alternatives. When calculating the results, we only used the training and validation data provided by the BioCreative organizers by splitting the training data into training and validation data and used the validation data as the test data.

Data augmentation

Table 2 shows the results from data augmentation for the four tracks. In general, the effect of augmentation depends on the specific NLP tasks and data sets. When calculating the results, we only used the training and validation data provided by the BioCreative organizers by splitting the training data into training and validation data and used the validation data as the test data.

Ensemble models

The majority vote approach gave a 0.753 F1 score on the test data for Track 1, which is substantially higher than individual models (increase by at least 1%). The averaging approach

Table 3. Track 2 chemical named entity recognition sub-task results. Our team (#128) ranked the second for this sub-task

File	Strict-P	Strict-R	Strict-F
Run 1	0.8544	0.8658	0.8600
Run 2	0.8643	0.8403	0.8521
Run 3	0.8440	0.7896	0.8159
Run 4	0.8457	0.8617	0.8536
Baseline	0.8440	0.7877	0.8149
Best (Team 139)	0.8759	0.8587	0.8672

Table 4. Track 2 chemical normalization sub-task results. Our team (#128) ranked second for this sub-task

File	Strict-P	Strict-R	Strict-F
Run 1	0.7833	0.8339	0.8078
Run 2	0.7792	0.8434	0.8101
Run 3	0.7780	0.8257	0.8011
Run 4	0.7755	0.8318	0.8027
Baseline	0.8151	0.7644	0.7889
Best (Team 110)	0.8621	0.7702	0.8136

Table 5. Track 2 chemical indexing sub-task results. Our team (#128) ranked first for this sub-task (unofficial). The official best performing team's result is also shown

File	Strict-P	Strict-R	Strict-F
Run 1	0.4424	0.5286	0.4817
Run 2	0.4397	0.5344	0.4825
Run 3	0.3776	0.3781	0.3779
Run 4	0.3805	0.3814	0.3809
Baseline	0.3134	0.6101	0.4141
Best (Team 110)	0.5351	0.4133	0.4664

gave 0.9081 F1 score on the test data compared to 0.9027 using the baseline BERT model for Track 5.

Detailed results for Track 2

Track 2 has three sub-tasks: named entity recognition, normalization and indexing. The result of named entity recognition is shown in Table 3, together with the baseline model performance and the performance of the best team according to Strict-F1 score. We (Team 128) ranked second according to Strict-F1 score. Strict-F1 score is the F1 score calculated when a prediction is considered as correct if the predicted entity overlaps strictly with the true entity, the same as the definition of Strict-P and Strict-R. The settings for different runs are as follows:

Run 1:

a. Data augmented by:

(1) replacing each of the chemical entities with a random string;

(2) selecting 50% of sentences which contain chemical entities and randomly choosing one nonchemical entity and replace it with random string while the chemical entities remain unchanged;

b. Using Ab3P to post-process the prediction results to add chemical entity tags and remove wrong chemical tags;

Run 2:

a. Data augmented by:

(1) replacing each of the chemical entities with a random string;

(2) selecting 70% of sentences which contain chemical entities and randomly choosing one nonchemical entity and replacing it with a random string while the chemical entities remain unchanged;

b. Same as Run 1;

c. Same as Run 1;

Run 3:

a. Same as Run 2;

b. Same as Run 1;

c. Same as Run 1;

d. Using the BioBERT protein NER model to detect protein entity and changing the label of the chemical entities which are part of a longer protein name to 'O' if they were labelled as 'B' or 'I';

Run 4:

a. Data augmented by:

(1) replacing each of the chemical entities with a random string;

(2) for all sentences which contain chemical entities and randomly selecting one nonchemical entity and replacing it with random string while the chemical entities remain unchanged;

b. Same as Run 1;

c. Same as Run 1;

d. Same as Run 3.

The result of chemical normalization is shown in Table 4. We also ranked second on this sub-task.

The results for chemical indexing sub-task are shown in Table 5 and our team performed the best for this sub-task. It is unofficial ranking because we were invited to submit our result in 1 month after the challenge has ended.

We performed error analysis by comparing BioBERT and PubMedBERT NER results. We found that (i) BioBERT tended to label nonchemical abbreviations as chemicals; (ii) BioBERT tended to label other entities related to chemical to chemical, such as diseases and viruses, and (iii) both BioBERT and PubMedBERT still make simple mistakes, indicating room for further improvements.

Detailed results for Track 3

The performance of our submissions for Track 3 is shown in Table 6 together with the baseline and best performances. The Overlapping F1 score, precision, and recall were calculated by considering a prediction as correct if the predicted entity partially overlaps with the true entity. Our team (#128) ranked fourth for this track. The settings for the three submissions are as follows: The first classifier (Submission 1) utilized the PubMedBERT (full-text) pre-trained model and fine-tuned with the BioCreative_TrainTask3.0, BioCreative_TrainTask 3.1 and their augmented data sets, plus SMM4H'18 data set. Data augmentation strategy is the first strategy, which is generating n copy of the original data set by replacing true entities with randomly chosen medication mentions where the medication mention pool is generated from only BioCreative_TrainTask3.0 and 3.1 data sets and $n = 1$. When $n > 1$, we repeated the process n times to generate n copies of data.

The second classifier (Submission 2) utilized the PubMedBERT (full-text) pre-trained model and fine-tuned with the BioCreative_TrainTask3.0, BioCreative_TrainTask 3.1 and

Table 6. Performances of submissions for Track 3

	Overlapping			Strict		
Submission	F1 score	Precision	Recall	F1 score	Precision	Recall
1	0.764	0.747	0.782	0.738	0.721	0.755
2	0.763	0.712	0.823	0.732	0.682	0.789
3	0.794	0.744	0.85	0.762	0.714	0.816
All participants (mean ± SD)	0.749 ± 0.0596	0.811	0.709	0.696 ± 0.072	0.754	0.658
Baseline	0.773	0.908	0.673	0.758	0.890	0.660
Best	0.838	0.832	0.844	0.804	0.799	0.810

their augmented data sets, plus SMM4H'18 data set. The augmented data sets are generated by two strategies: the first one is generating n copies of the original data by replacing true entities with randomly chosen medication mentions where the medication mention pool is generated from BioCreative_TrainTask3.0 and 3.1 and SMM4H'18 three data sets and $n = 10$. The second one is generating n copies of the original data by dropping a randomly selected word which is not or not belong to a true entity, where $n = 1$.

The third classifier (Submission 3) utilized the PubMedBERT (full-text) pre-trained model and fine-tuned with the BioCreative_TrainTask3.0, BioCreative_TrainTask 3.1 and their augmented data sets, plus SMM4H'18 data set. The augmented data sets were generated by two strategies: the first one is generating n copies of the original data by replacing true entities with randomly chosen medication mentions where the medication mention pool is generated from only BioCreative_TrainTask3.0 and 3.1 data sets and $n = 3$. The second one is generating n copies of the original data by dropping a randomly selected word which is not or belong to a true entity, where $n = 1$.

We performed error analysis for three different augmentation strategies we used in this track: (i) dropping a random word, which is not a true entity; (ii) replacing a true entity with random strings; and (iii) replacing a true entity with another word of the same entity type. We found that the third type of augmentation strategy performed better in general. When replacing a true entity with random strings, sometimes the model will predict a non-entity abbreviation as an entity. This happens likely because most of the random strings we created look like abbreviations. When dropping a random word, the model makes errors that correspond to not understanding the context of an entity well, likely because we did not generate enough variations for the entities to force the model to learn the context. But overall, all these three types of augmentation suffer from making simple mistakes occasionally, indicating that there are still enough room for further improvement. The reasons for the simple mistakes are not very clear.

Conclusion and discussion

In this paper, we described the methods we have used in BioCreative Challenge VII. As mentioned briefly in Introduction, we have some key components for achieving good performance in the BioCreative Challenge tasks: (i) pre-trained NLP models; (ii) data augmentation and (iii) ensemble modelling. Below we give some detailed discussions on these components.

First, it is important to test all the available pre-trained models. We have found that PubMedBERT worked better than other models in our own study.

Second, we have found that data augmentation methods are helpful in most of the cases. However, there are many different data augmentation strategies. Data augmentation may also introduce some biases or noise in the train data. In retrospect, we did not fully explore this method in our study in some tracks. It is worth further investigation in the future, especially in cases where only limited data are available.

Third, we have used ensemble models for two of the tracks, and they were both helpful. It would be more beneficial if we used ensemble models for all the tracks. It may also help if we could explore more options in ensemble models. Currently, we found that majority voting gave the best performance, probably due to its robustness. We expect that training additional machine learning model may give even better results at least for some tasks. In addition to different pre-trained models and parameter settings, different random seeds and different check points can also generate different models that can be used in ensemble. Given all these options, finding the best strategy to combine them would be an interesting topic for future studies.

It is worth mentioning that some of the above strategies we have discussed may be only important for winning competitions or for cases where a small improvement in performance may bring large benefits in practice. For cases where small improvements are not very important, a good baseline model with optimized parameters usually gives quite satisfactory results. For many real applications, ensemble models are likely not practical due to computational concerns.

One final note we would like to mention is that we have tried many different ideas during the whole summer of 2021, but most of them failed to improve the baseline models. This observation showed, on one hand, that the baseline models built from pre-trained models have achieved quite good performance. On the other hand, it indicates that the pre-trained models have learned or incorporated a great deal of information that we probably do not explicitly know. The additional gain we aimed to achieve through the ideas we have tried had probably already been achieved by the pre-trained models in some implicit ways.

Funding

National Institute of General Medical Science of National Institutes of Health (R01GM126558 to J.Z., in part). The funder had no role in the study design, data collection and analysis, decision to publish or preparation of the manuscript.

Conflict of interest

None declared.

References

1. Rzhetsky,A., Seringhaus,M. and Gerstein,M. (2008) Seeking a new biology through text mining. *Cell*, **134**, 9–13. 10.1016/j.cell.2008.06.029.
2. Leitner,F., Krallinger,M., Rodriguez-Penagos,C. *et al.* (2008) Introducing meta-services for biomedical information extraction. *Genome Biol.*, **9 Suppl 2**, S6. 10.1186/gb-2008-9-s2-s6.
3. Pyysalo,S., Ginter,F., Heimonen,J. *et al.* (2007) BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinform.*, **8**, 50. 10.1186/1471-2105-8-50.
4. Chowdhary,R., Zhang,J., Tan,S.L. *et al.* (2013) PIMiner: a web tool for extraction of protein interactions from biomedical literature. *Int. J. Data Min. Bioinform.*, **7**, 450–462. 10.1504/IJDMB.2013.054232.
5. Chowdhary,R., Tan,S.L., Zhang,J. *et al.* (2012) Context-specific protein network miner—an online system for exploring context-specific protein interaction networks from the literature. *PLoS One*, **7**, e34480. 10.1371/journal.pone.0034480.
6. Balaji,S., McClendon,C., Chowdhary,R. *et al.* (2012) IMID: integrated molecular interaction database. *Bioinformatics*, **28**, 747–749. 10.1093/bioinformatics/bts010.
7. Bell,L., Chowdhary,R., Liu,J.S. *et al.* (2011) Integrated bio-entity network: a system for biological knowledge discovery. *PLoS One*, **6**, e21474. 10.1371/journal.pone.0021474.
8. Chowdhary,R., Zhang,J. and Liu,J.S. (2009) Bayesian inference of protein-protein interactions from biological literature. *Bioinformatics*, **25**, 1536–1542. 10.1093/bioinformatics/btp245.
9. Qu,J., Steppi,A., Zhong,D. *et al.* (2020) Triage of documents containing protein interactions affected by mutations using an NLP based machine learning approach. *BMC Genom.*, **21**, 773. 10.1186/s12864-020-07185-7.
10. Lung,P.-Y., He,Z., Zhao,T. *et al.* (2019) Extracting chemical-protein interactions from literature using sentence structure analysis and feature engineering. *Database (Oxford)*, bay138. 10.1093/database/bay138.
11. Yu,K., Zhao,T., Zhao,P. *et al.* (2017) Extraction of protein-protein interactions using natural language processing based pattern matching. In: *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Kansas City, MO, pp. 1292–1295.
12. Qu,J., Steppi,A., Hao,J. *et al.* (2017) Mining protein interactions affected by mutations using a NLP based machine learning approach. In: *Proceedings of BioCreative VI Workshop*, Kansas City, MO, pp. 130–135.
13. Lung,P.-Y., Zhao,T., He,Z. *et al.* (2017) Extracting chemical-protein interactions from literature. In: *Proceedings of BioCreative VI Workshop*, Kansas City, MO, 160163.
14. Huang,C.C. and Lu,Z. (2016) Community challenges in biomedical text mining over 10 years: success, failure and the future. *Brief. Bioinformatics*, **17**, 132–144. 10.1093/bib/bbv024.
15. Arighi,C.N., Lu,Z., Krallinger,M. *et al.* (2011) Overview of the BioCreative III workshop. *BMC Bioinform.*, **12**, S1. 10.1186/1471-2105-12-S8-S1.
16. Krallinger,M., Morgan,A., Smith,L. *et al.* (2008) Evaluation of text-mining systems for biology: overview of the Second BioCreative community challenge. *Genome Biol.*, **9 Suppl 2**, S1. 10.1186/gb-2008-9-s2-s1.
17. Hirschman,L., Yeh,A., Blaschke,C. *et al.* (2005) Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinform.*, **6 Suppl 1**, S1. 10.1186/1471-2105-6-S1-S1.
18. *LitCoin Natural Language Processing (NLP) Challenge*. (2021) https://ncats.nih.gov/funding/challenges/litcoin (10 August 2022, date last accessed).
19. Miranda,A., Mehryary,F., Luoma,J. *et al.* (2021) Overview of DrugProt BioCreative VII track: quality evaluation and large scale text mining of drug-gene/protein relations. In: *Proceedings of the BioCreative VII Challenge Evaluation Workshop*, Virtual meeting, 11.
20. Leaman,R., Islamaj,R. and Lu,Z. (2021) The overview of the NLM-Chem BioCreative VII track full-text chemical identification and indexing in PubMed articles. In: *Proceedings of the BioCreative VII Challenge Evaluation Workshop*, Virtual meeting, 108.
21. Weissenbacher,D., O'Connor,K., Rawal,S. *et al.* (2021) VII - Task 3: automatic extraction of medication names in tweets. In: *Proceedings of the BioCreative VII Challenge Evaluation Workshop*, Virtual meeting, 163.
22. Chen,Q., Allot,A., Leaman,R. *et al.* (2021) Overview of the BioCreative VII LitCovid track: multi-label topic classification for COVID-19 literature annotation. In: *Proceedings of the BioCreative VII Challenge Evaluation Workshop*, Virtual meeting, 266.
23. Islamaj,R., Leaman,R., Kim,S. *et al.* (2021) NLM-Chem, a new resource for chemical entity recognition in PubMed full text literature. *Sci. Data*, **8**. 10.1038/s41597-021-00875-1.
24. Islamaj,R., Leaman,R., Cissel,D. *et al.* (2021) The chemical corpus of the NLM-Chem BioCreative VII track full-text chemical identification and indexing in PubMed articles.
25. Chen,Q., Allot,A. and Lu,Z. (2020) Keep up with the latest coronavirus research. *Nature*, **579**, 193–193. 10.1038/d41586-020-00694-1.
26. Chen,Q., Allot,A. and Lu,Z. (2021) LitCovid: an open database of COVID-19 literature. *Nucleic Acids Res.*, **49**, D1534–D1540. 10.1093/nar/gkaa952.
27. Devlin,J., Chang,M.W., Lee,K. *et al.* (2018) Pre-training of deep bidirectional transformers for language understanding. preprint arXiv:1810.04805.
28. Lee,J., Yoon,W., Kim,S. *et al.* (2019) Biobert: pre-trained biomedical language representation model for biomedical text mining. preprint arXiv:1901.08746.
29. Gu,Y., Tinn,R., Cheng,H. *et al.* (2020) Domain-specific language model pretraining for biomedical natural language processing. arXiv **abs/2007.15779**.
30. Reimers,N. and Gurevych,I. (2019) Sentence-BERT: sentence embeddings using Siamese BERT-networks. arXiv **abs/1908.10084**.
31. Raffel,C., Shazeer,N.M., Roberts,A. *et al.* (2020) Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv **abs/1910.10683**.
32. Peng,Y., Yan,S. and Lu,Z. (2019) Transfer learning in biomedical natural language processing: an evaluation of BERT and ELMo on ten benchmarking datasets. BioNLP@ACL.
33. Beltagy,I., Lo,K. and Cohan,A. (2019) SciBERT: a pretrained language model for scientific text. 3615–3620.
34. Liu,Y., Ott,M., Goyal,N. *et al.* (2019) RoBERTa: a robustly optimized BERT pretraining approach. arXiv **abs/1907.11692**.
35. Alsentzer,E., Murphy,J., Boag,W. *et al.* (2019) Publicly available clinical BERT embeddings. 72–78.
36. Erdengasileng,A., Keqiao,L., Han,Q. *et al.* (2021) A BERT-based hybrid system for chemical identification and indexing in full-text articles. BIORXIV/2021/466183.
37. Sohn,S., Comeau,D.C., Kim,W. *et al.* (2008) Abbreviation definition identification based on automatic precision estimates. *BMC Bioinform.*, **9**, 402. 10.1186/1471-2105-9-402.
38. Wei,C.H., Allot,A., Leaman,R. *et al.* (2019) PubTator central: automated concept annotation for biomedical full text articles. *Nucleic Acids Res.*, **47**, W587–W593. 10.1093/nar/gkz389.