applied sciences

MDPI

Article

Biomedical Text NER Tagging Tool with Web Interface for Generating BERT-Based Fine-Tuning Dataset

Yeon-Ji Park [1], Min-a Lee [1], Geun-Je Yang [1], Soo Jun Park [2,*] and Chae-Bong Sohn [1,*]

[1] Department of Electronics and Communications Engineering, Kwangwoon University, Seoul 01897, Republic of Korea
[2] Welfare & Medical ICT Research Department, Electronics and Telecommunications Research Institute, Daejeon 34129, Republic of Korea
* Correspondence: psj@etri.re.kr (S.J.P.); cbsohn@kw.ac.kr (C.-B.S.)

Abstract: In this paper, a tagging tool is developed to streamline the process of locating tags for each term and manually selecting the target term. It directly extracts the terms to be tagged from sentences and displays it to the user. It also increases tagging efficiency by allowing users to reflect candidate categories in untagged terms. It is based on annotations automatically generated using machine learning. Subsequently, this architecture is fine-tuned using Bidirectional Encoder Representations from Transformers (BERT) to enable the tagging of terms that cannot be captured using Named-Entity Recognition (NER). The tagged text data extracted using the proposed tagging tool can be used as an additional training dataset. The tagging tool, which receives and saves new NE annotation input online, is added to the NER and RE web interfaces using BERT. Annotation information downloaded by the user includes the category (e.g., diseases, genes/proteins) and the list of words associated to the named entity selected by the user. The results reveal that the RE and NER results are improved using the proposed web service by collecting more NE annotation data and fine-tuning the model using generated datasets. Our application programming interfaces and demonstrations are available to the public at via the website link provided in this paper.

Keywords: dataset generation; BERT; tagging tool; web service; natural language process; text mining; named-entity recognition; fine-tuning model

check for updates

Citation: Park, Y.-J.; Lee, M.-a.; Yang, G.-J.; Park, S.J.; Sohn, C.-B. Biomedical Text NER Tagging Tool with Web Interface for Generating BERT-Based Fine-Tuning Dataset. *Appl. Sci.* **2022**, *12*, 12012. https://doi.org/10.3390/app122312012

Academic Editors: Juan A. Gómez-Pulido and Alexander N. Pisarchik

Received: 14 September 2022
Accepted: 22 November 2022
Published: 24 November 2022

1. Introduction

Recently, text mining has garnered significant academic attention [1,2] and various web-based annotation tools have been proposed [3–18]. In particular, text mining technologies have evolved to provide automatically generated pre-annotations using machine learning, increasing the amount of data curation to unprecedented levels. However, the amount of generated training data remains limited; only data from general fields are available, rather than those on specialized topics such as diseases and genes [19–22]. A tagging tool called ezTag, which annotates biological parameters, was developed to address this shortcoming [23], but it lacks adequate project management as well as support for collective annotation by multiple users. To overcome these limitations, a tagging tool called TeamTat was proposed [24], which is a web-based annotation tool that allows multiple users to manage annotation projects together. Moreover, the comments of independent users can be collected and viewed simultaneously.

In our study, the BioBERT model, which integrates biomedical text corpora with bidirectional encoder presentations from transformers (BERT) models, is employed to construct a web system [25]. In the tasks of named-entity recognition (NER), relation extraction (RE), and question answering (QA), the BioBERT model outperforms current state-of-the-art models. Consequently, our approach surpasses earlier text mining tools in terms of entity tagging. In addition to delivering entity tagging options comparable to those supplied by

other text mining systems, our system enables and reflects users' ability to tag individual entities themselves. Users no longer need to spend substantial time and effort on entity tagging with our system. Furthermore, we enable the collection of huge volumes of training datasets in specific domains (biomedical domains) as opposed to general domains, hence substantially increasing the quantity of data curation.

In this paper, we integrate a tagging tool into NER and RE web interfaces using BERT. Moreover, we improve the quality of NER results by fine-tuning the deep learning model. The system architecture of the proposed tagging tool is depicted in Figure 1. First, the user inputs the new NE annotation, comprising a list of words and categories (e.g., diseases, genes/proteins) of named entities selected by the user, via the web. Second, in the NE annotation collection, users can download and collect new NE annotations. The new NE annotation collection includes a list of words and categories (e.g., disease, genes/proteins) of named entity selected by the user. After the amount of input NE annotation data reaches a certain threshold, users can fine-tune the deep learning model to create a new pretrained model. This allows the system to yield better NER results. Our main contributions are as follows:

- Our web service is a biomedical text mining tool that uses deep learning based high-performance BioBERT NER and RE models;
- The NER result is displayed in a unique color, and the RE result is shown in a graph;
- We introduced a tagging tool system to our service so that users can tag the entities they desired;
- Our web service can download newly tagged annotations and use it as a dataset for retraining. If this is used for retraining, better NER results can be used through the new pretrained model;
- This service is freely available at http://nertag.kw.ac.kr (accessed on 10 October 2022)

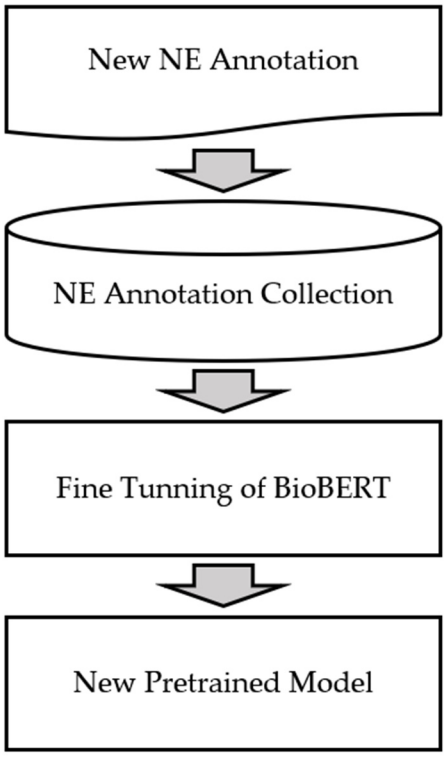

Figure 1. Structure of tagging tool.

2. Related Work

Manual annotation is labor-intensive and time-consuming. The tagging tool was developed to automate the manual selection of target terms, especially when the number of tags

is very large, as well as the search for and selection of tags corresponding to each term. Tagging tasks can be performed more efficiently by extracting pre-tagged terms from sentences, presenting them to users, and selecting and providing tags applicable to those terms.

2.1. Textpresso Central(TPC)

Recently, text mining—the process of extracting high-quality information from text—has become quite popular. A tagging tool was developed to make annotations more convenient; however, it exhibited poor accuracy in the biological field since the original tagging tools were only trained in general-purpose languages. Textpresso Central (TPC) addresses this shortcoming and enables customized text mining for biomedical researchers. Figure 2 presents an overview of the pipeline of Textpresso Central. First, the original file is tokenized, and the entire text string is identified in the first stage common analysis structure (CAS) files. Next, the lexical annotator reads the generated CAS file, identifies vocabulary items, labels each category, annotates the location within each category, and records these annotations in the second stage CAS files.

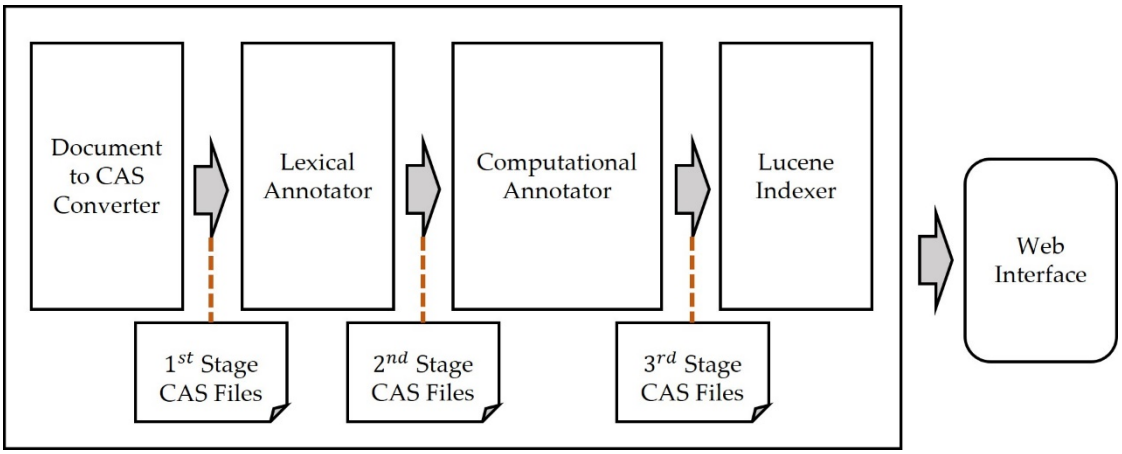

Figure 2. Pipeline of Textpresso Central.

Subsequently, the computational annotator runs the second stage CAS files, and the resulting annotation is added and recorded in the third stage CAS files. The indexer indexes all annotations and keywords and adds them to the third stage CAS files to make it searchable on the web. The final processed files can be used in text mining and machine learning algorithms.

2.2. ezTag

Existing tagging tools focus primarily on common categories, such as diseases and genes, because of the limitations of available training data. ezTag is a web-based annotation tool that allows curators to annotate various biological concepts and produce manual training data, irrespective of the presence or absence of existing training data, on the basis of pretrained tags such as GNormPlus [26], tmVar [27], and TaggerOne [28]. It is interactive and allows manual editing of automatically tagged text to generate new annotation data for improved model training. It is capable of both automatic and manual annotation, with the former performed using a machine learning-based tagger, a pretrained tagger, and a string match-based tagger. Based on this diversity, users of ezTag can perform interactive learning on adaptive identity tagging. The machine learning-based tagger normalizes and recognizes jointly named entities, and the pretrained tagger assigns each entity one of six tags: disease, gene/protein, chemical, sequence, organism/species, and variations. Finally, the string matching-based tagger identifies bio-entities and uses user-supplied words for concept ID assignment.

Figure 3 presents an overview of the ezTag system. The entire repository of PubMed-Central (PMC) [29] and PubMed abstracts is preprocessed into BioC documents, which

are accessed via RESTful API. This enables the processing and sharing of BioC documents. BioC is a simple XML-based format for sharing useful text documents and annotations. Lexicons have two roles: assigning conceptual IDs for machine learning-based taggers, and string matching-based tagging. The pretrained tagger, string matching-based tagger, and machine learning-based tagger are utilized to finally output the annotated BioC documents.

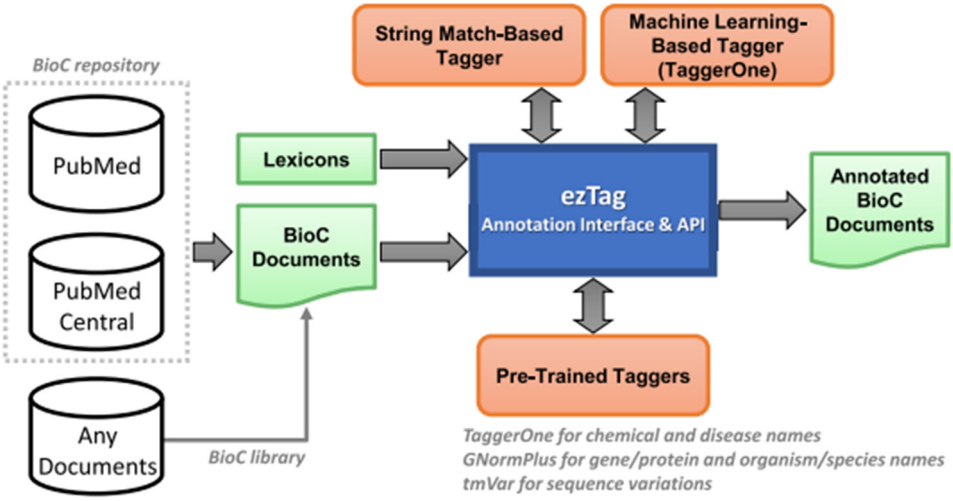

Figure 3. Overview of the ezTag system (cited from [23]).

2.3. TeamTat

Despite its many advantages, e.g., automatic annotation, ezTag does not support multi-user annotation, picture display, or adequate project management. These features are implemented in TeamTat, which is a web-based annotation tool that can efficiently manage team annotation projects, focusing primarily on project management. TeamTat is capable of accepting textual documents in various formats as input: text, PDF, and BioC. It also implements an intuitive interface for all users to review and analyze common annotations individually, supports a simple format for sharing data with annotations generated via text mining [1,2], and displays full texts showing the entire document, including figures, which are essential parts of biomedical annotation and curation.

TeamTat allows project managers to set up projects, select documents to be annotated, select annotators, and distribute documents to annotators. Annotation rounds comprise multiple annotation teams working independently. Members of each team may review the comments in case of disagreement between fellow annotators. To avoid bias in annotations, TeamTat uses an anonymity-based method. At the end of every round, the project manager calculates the consensus statistics between the annotators and decides whether to finalize the corpus or continue the task. Annotations can be tracked over all annotation rounds, and data can be downloaded at any time during annotation processing.

3. Materials and Methods

The NER and RE web interfaces are web pages showing results of BioBERT NER and RE. The proposed system incorporates the NER dataset generation module of the tagging tool system, as depicted in Figure 4, into the existing web interface of NER and RE with BERT.

PubMed identifier (PMID) received based on the user's input is used to extract information such as title, abstract, and authors using the Entrez-NCBI search engine. The BERT-based training model performs data normalization and outputs the NER and RE results for each category.

BERT-based model: BERT was released by Google in 2017 and is based on a transformer and a deep learning model comprising encoder and decoder structures [30,31]. The transformer preprocesses the input data, transmits it to the encoder, and then inputs the data

into the decoder. A padding mask is applied to the encoder data to reduce the amount of computation required. Next, a look-ahead mask is applied to the decoder data to hide the prediction word in advance. Then, the padding mask is applied to the data and the existing input data are used as inputs to the encoder. The decoder receives the decoder input data, the result calculated by the encoder, and the look-ahead mask and padding mask values as inputs. Finally, the following word is predicted based on the value calculated by the decoder.

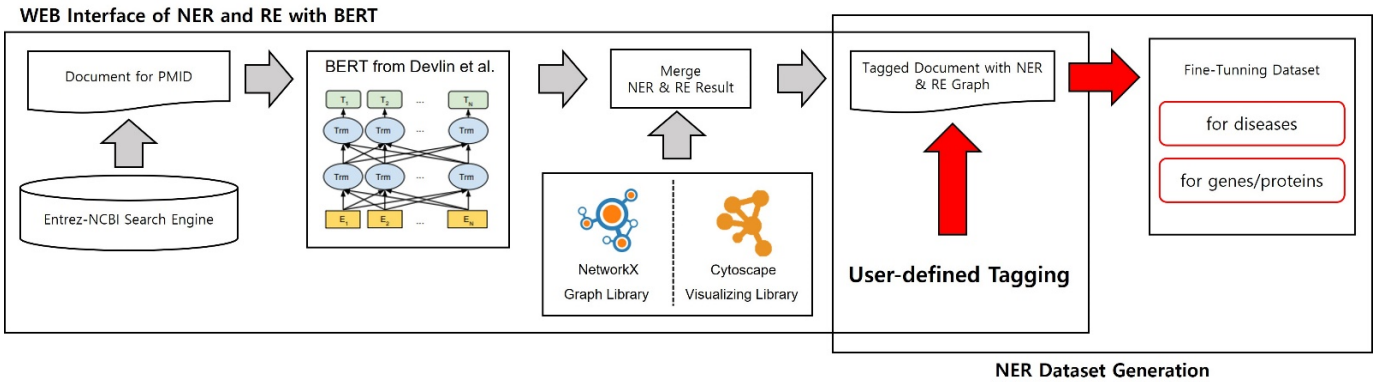

Figure 4. Overview of the proposed web interface of NER and RE with BERT.

Figure 5 depicts the encoder and decoder structures within the transformer. We focus solely on the encoder as BERT is trained using only this component of the transformer. Both the encoder and the decoder are responsible for three main tasks: positional encoding, multi-head self-attention, and position-wise feed-forward networks.

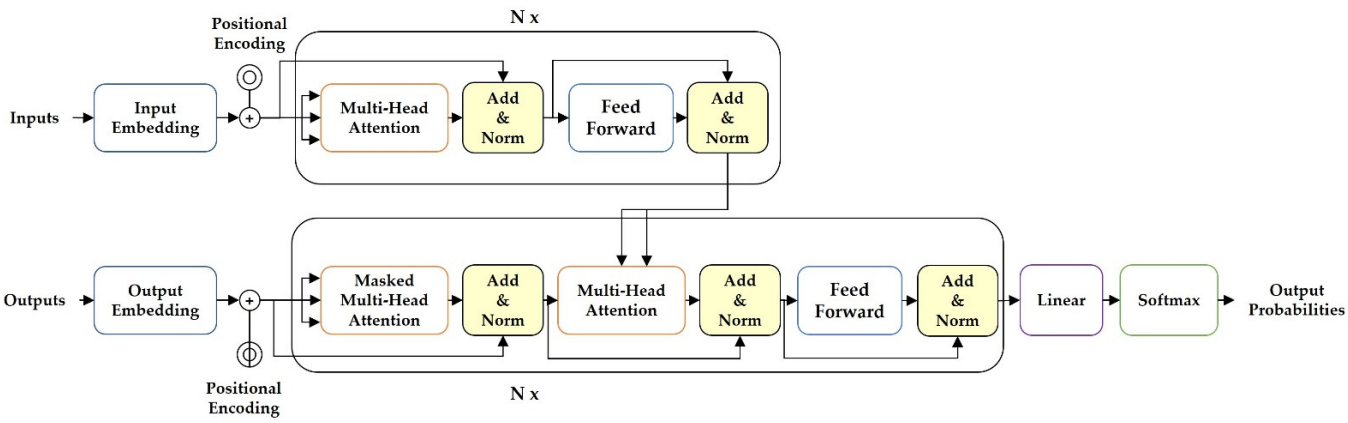

Figure 5. Structure of the transformer.

Positional encoding considers positional information of words as word order is essential to comprehension of sentence structures in language. By incorporating positional encoding of the same dimension as the existing embedding, embeddings with time signals can be used as input. Attention can be classified as scaled dot-product attention or multi-head attention. Multi-head attention yields better values by concatenating multiple scaled dot-product attentions. Multi-head attention is used because it reduces the size of vectors and enables parallel processing when multiple attention functions are used.

$$\mathrm{MultiHead}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \mathrm{Concat}(\mathrm{head}_1,\ldots,\mathrm{head}_\mathrm{h})\mathrm{W}^\mathrm{O} \tag{1}$$

$$\mathrm{head}_\mathrm{i} = \mathrm{Attention}\left(\mathrm{QW_i}^\mathrm{Q},\mathrm{Ki}^\mathrm{K},\mathrm{VW_i}^\mathrm{V}\right) \tag{2}$$

The aforementioned equations give the formula for multi-head attention. The attention of each head (keys, values, queries) is divided by h to reduce the dimension and concatenated. Finally, position-wise feed-forward networks are layers that extract information

from sentences. They are implemented alongside an attention layer and are composed of two linear transformations, using ReLU as an activation function.

Generative Pretrained Transformer (GPT) is another model based on the decoder of the aforementioned transformer. It exhibits weak contextual understanding because it only reads sentences from left to right, i.e., in one direction. BERT, which reads text bidirectionally using only the encoder of the aforementioned transformer, can understand the context more naturally.

Figure 6 depicts input representation of BERT. Two tokens stand out in the input section: [CLS] and [SEP]. The first token of every sentence is [CLS]; the classification value determined by BERT is dependent on the operation of this token. The [SEP] token is used at the end of each sentence to distinguish it from the subsequent one. BERT consists of three embedding layers: token embedding layer (word-piece embedding layer), segment embedding layer, and position embedding layer.

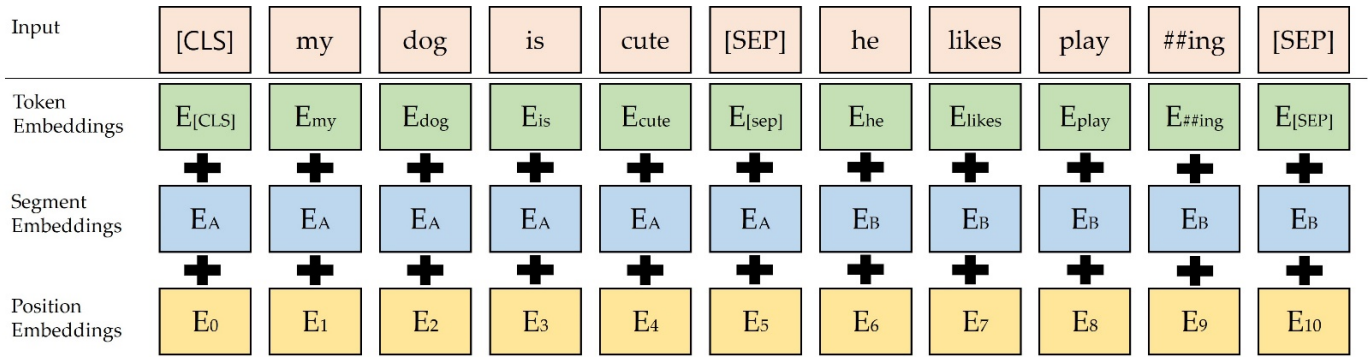

Figure 6. Input embedding of BERT (cited from [30]).

Word-piece embedding involves the embedding of a word that is an actual input, and is used to divide a sentence into token units. It enables more effective discrimination of tokens compared to simply separating tokens using spaces. For example, the word "playing" can be considered to be a token comprising the components, "play" and "ing". This enables the clear comprehension of each token by the deep learning model; thus, even when a misspelt or new word is used as input, the deep learning model performs satisfactorily as the words are divided into components that may have been trained previously. Segment embedding informs the deep learning model of the existence of two different sentences. When two sentences are input, they are distinguished by assigning a distinct number to each sentence. Position embedding encodes the relative location information of tokens. The deep learning model uses position embedding to determine the order of tokens. In general, position embedding uses the sine and cosine functions to create a matrix whose elements comprise word vectors depending on their positions. However, in BERT, positional information is obtained based on training, rather than the use of sine and cosine functions.

BioBERT Fine-tuning: BERT normally uses pretrained word embeddings in text mining tasks, but BioBERT creates a new entity by training word-piece embedding by itself during the fine-tuning and pretraining processes. However, if the word included in the existing text does not exist in the vocabulary of the embedding, providing alternative expressions for the corresponding word is more difficult that when it exists in the vocabulary of the embedding.

To overcome this, BioBERT employs word-piece embedding used in BERT to represent each word by dividing it into lower-level words. Since it can elicit more meaning, it is easy to deduce the meaning of a term from low-level words, even if there are unseen words or misspellings.

To utilize this advantage properly, the pretrained weights are adjusted using word-piece embedding while fine-tuning BioBERT.

$$p(y_i = k|T_i) = softmax(T_i W^{\mathsf{T}} + b)_k, \;\; k = 0, 1, \ldots, 6 \tag{3}$$

Here, T_i denotes the last hidden size corresponding to the token, i; p denotes the label probability; b denotes bias; W denotes the classification layer; and $k = 7$. The equation for classification loss, L, is as follows:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\log(p(y_i|T_i;\theta)) \tag{4}$$

where N denotes the sequence length and θ denotes the trainable parameter.

BioBERT NER is a text mining task that extracts domain-specific entities via directly trained word-piece embedding. Token classification, which assigns labels to word tokens, is performed using a single output layer. In the BERT embedding process, the meanings of even low-level words are extracted to facilitate the inference of meanings of input words that are not included in the embedding vocabulary. RE classifies the relation between entities in a biomedical corpus. Entity identification in the NER model must be performed before processing RE. The RE task, which follows the NER task, identifies all possible entity pairs in the sentence and classifies the relation. For instance, consider a sentence in which the words disease category A, B, and gene/protein category C are extracted from the NER task. In such a case, the pair of relationships between the two categories will be (A, C), (B, C). The two relation pairs are substituted with a specific mask to generate two relation sentences that serve as input. The outcomes of NER and RE fine-tuning are visually given to the user. NetworkX was utilized to convert RE data to graphs, and the Cytoscape module was utilized to visualize graph data for interaction. A more detailed description is offered in the result section.

NER Dataset Generation for Retraining: The existing web service visually displays the NER results and RE graph corresponding to the desired PMID. However, it does not reflect the tagging information defined by the user, which may be different in various contexts. The proposed system, on the other hand, is configured to reflect user-defined tagging information, and it can be downloaded and utilized in the form of a retraining dataset. The produced dataset is provided in a format that allows the user to apply it straight to the model without modification. The detailed procedure for collecting a dataset for retraining is as follows.

We show the user the results of the BioBERT NER fine-tuning in the form of a clickable button in text format. The user is then able to click on the desired term to reflect the user-defined tagging information. By clicking the "save as tsv" button, it is possible to download data that contain all needed tagging information.

The data are then categorized as genes/proteins or diseases and saved in the form of a tab separated value (tsv) file. The tsv file format is a tab-delimited data table with data. It is configured for simple application to training based on the tsv file format used in training. In addition, the data were tagged through the tagging technique used primarily by tagging systems. The tagging system was introduced to integrate and recognize various tokens into named entities, and can be classified into two types: BIO tagging and BIOES tagging. BIO denotes Begin, Inside, and Outside. In BIO tagging, the token at the beginning of a named entity is denoted by B, tokens in the middle of the object name are denoted by I, and tokens that are not named entities are denoted by O. BIOES denotes Begin, Inside, Outside, End, and Single. In BIOES tagging, the B, I, and O tags are identical to the ones in BIO tagging. Additionally, when the named entity is a single token, it is denoted by S, and when the named entity comprises three or more tokens, it is denoted by E. The dataset generation module is fine-tuned using BIO tagging, which is used in BERT.

4. Results

In this section, we discuss the following aspects of the proposed system:

- Implementation Environments;
- Web Service Implementation;
- Model Retraining.

Section 4.1 describes the system environment for implementation, Section 4.2 presents the implementation results obtained by adding the tagging tool and dataset generation function to the existing BERT-based web service, and Section 4.3 describes the result of retraining the generated dataset using the fine-tuned model.

4.1. Implementation Environments

Table 1 lists the details of the system environments used in the experiment in this paper. Web service implementation and model retraining are performed in identical system environments.

Table 1. System Environments.

System Environments	
CPU	Inter® Core™ i9-10920X 12-core Processor
RAM	96 GB
VGA	NVIDIA GeForce RTX 3090
OS	Windows 10
TOOL	Python 3.6.9

For our tagging tool, the NCBI Disease [22] and BC2GM [32] datasets were used in the NER task, and the GAD [33] dataset was used in the RE task. Table 2 depicts the NER model's Precision, Recall, and F1-score values. Here, it is evident that the BioBERT model we used has superior performance than the NER model used in other text mining tools. For this reason, we used the BioBERT model with the highest performance in the disease and gene/protein domains. Given that there is no existing BERT-based tagging tool, the purpose of this paper is not to improve performance, but rather to propose a tagging tool based on the highest-performing BioBERT model. There are no other text mining tools that show results for RE. Moreover, the purpose of RE is to let users locate NER results more precisely and conveniently. Therefore, it is difficult to compare our system's performance to the RE model utilized by other tagging tools, so our approach is unique.

Table 2. Performance comparison of NER models for genes/proteins, diseases.

Entity Types	Pretrained NER Models	Precision	Recall	F1-Score
Disease	BioBERT [25]	0.8904	0.8969	0.8936
	Sachan et al. [34]	0.8641	0.8831	0.8734
	CollaboNet [35]	0.8548	0.8727	0.8636
	LSTM-CRF (iii) of Habibi et al. [36]	0.8531	0.8358	0.8444
Gene/Protein	BioBERT [25]	0.8516	0.8365	0.8440
	Sachan et al. [34]	0.8181	0.8157	0.8169
	CollaboNet [35]	0.8049	0.7899	0.7973
	LSTM-CRF (iii) of Habibi et al. [36]	0.7750	0.7813	0.7782

4.2. Web Service Implementation

We now present the implementation results of the data generation module, including the proposed tagging tool, on existing web pages using BERT. The python-based Django web framework is used, and all web pages are implemented to be HTML/CSS compatible. Figure 7 depicts the existing system's interfaces. To receive the user-defined NE annotation information, the user interface for the tagging function depicted in Figure 8 is added to the NER part in Figure 7. All word tokens are converted into individual buttons to enable user interaction, enabling manual selection of desired words and phrases. When the user clicks on the beginning and end of a phrase, the intervening portions are automatically selected,

reducing the need to click on each phrase. The sample article belongs to Alzheimer's disease research, and its PMID is 26707559.

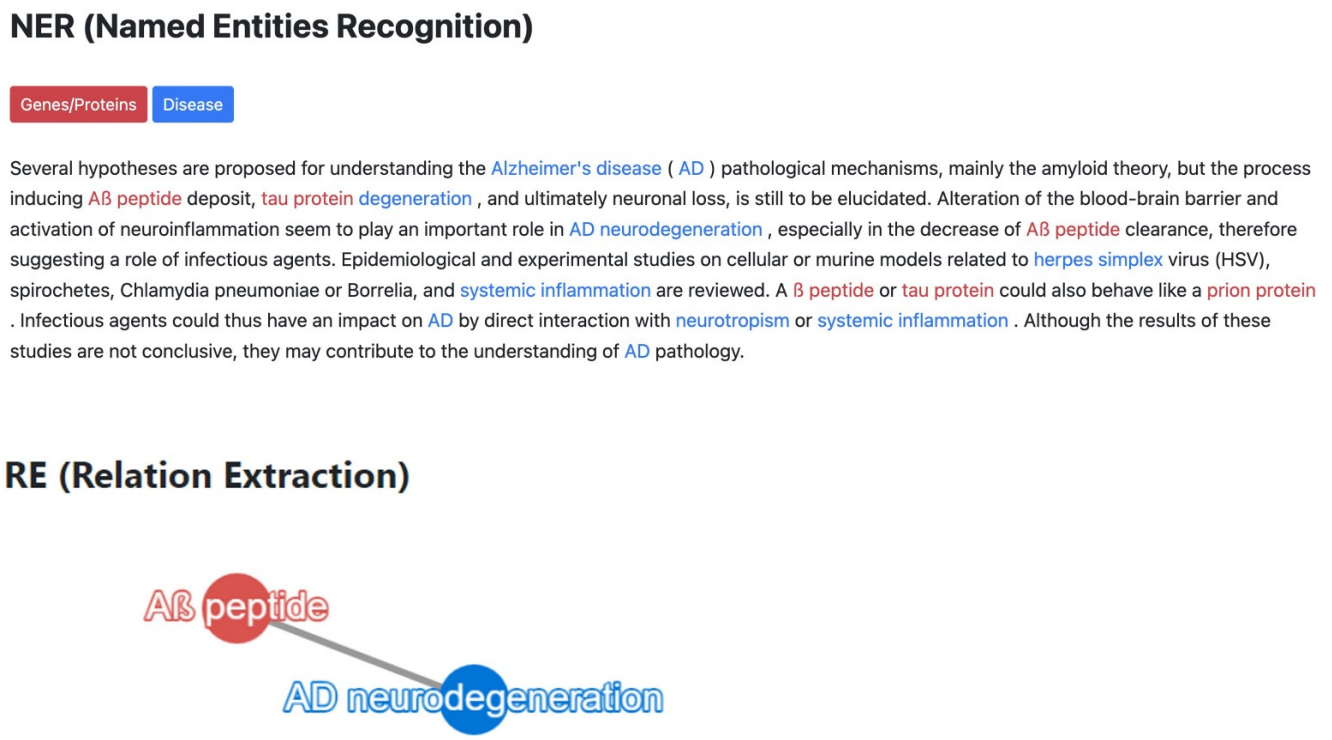

Figure 7. Original Web Interface of NER and RE with BERT.

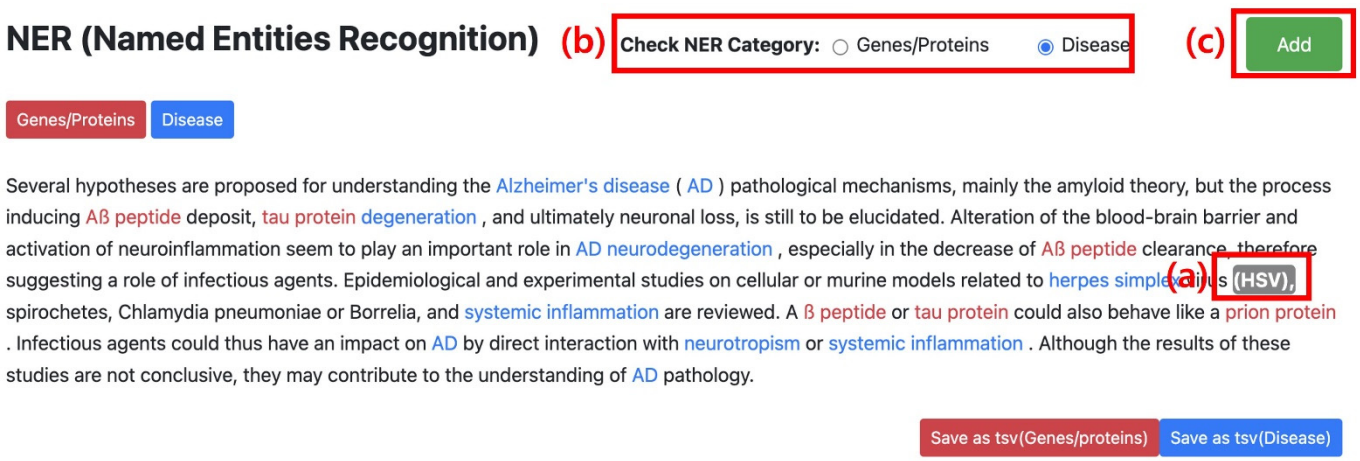

Figure 8. Proposed Web Interface of NER and RE with BERT using tagging tool system. (**a**) It is a word chosen by the user to form NE. (**b**) It is a user interface for categorizing the selected term. (**c**) It is a category-adding button.

New NE annotations are generated using the tagging tool as follows. To add a new annotation to textual data in the NER results, users must click the beginning and end of the desired word or phrase. Figure 8a depicts a phrase selected by a user that belongs to the disease category but has not been as such by BERT. In this case, the user can select the category to be applied by clicking "Check NER Category", as depicted in Figure 8b, and then clicking the "Add" button, as depicted in Figure 8c. This changes the color of the original tagged phrase, as illustrated in Figure 9.

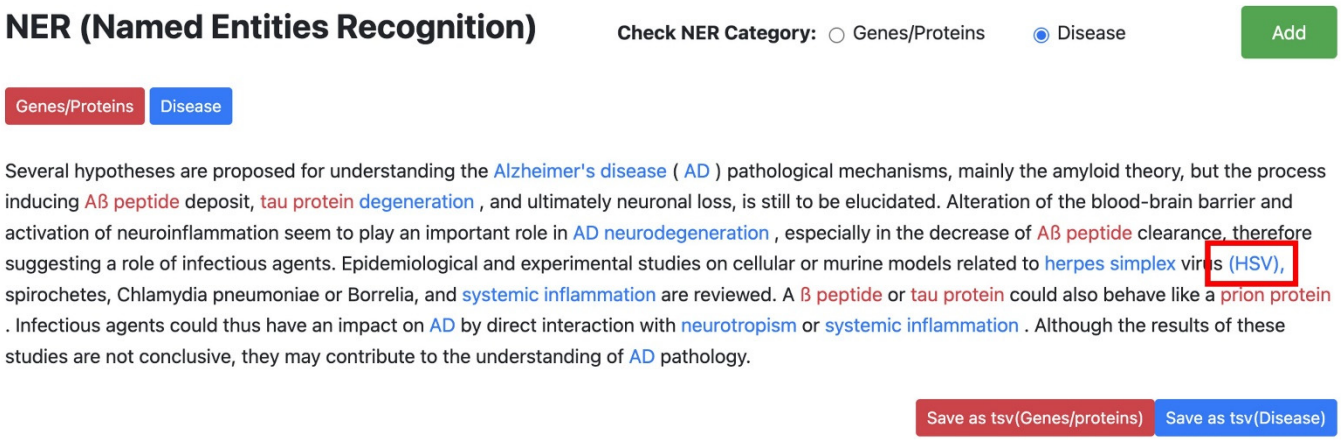

Figure 9. User-defined NE annotation information added to the information. Red box highlights terms that the user has manually annotated.

The added NE annotation information is applied in red for the genes/proteins category and in blue for the disease category, just as the existing tagged information is displayed. This change not only informs the user that a new annotation has been added, but also generates a new fine-tuning dataset by gathering manually annotated phrases. This dataset can be used to retrain the model.

4.3. Model Retraining

The new dataset created using the implemented web service can be used for model retraining to increase user-customized inference accuracy. Because there may be missing results in the previously learned dataset, our tagging tool enables users to generate a more accurate dataset. There are three ways to retrain using a pretrained model: retraining the entire model, retraining some layers and classifiers while retaining a specific frozen layer, and retraining only the classifier while retaining the entire layer. Since the dataset added by the user includes word tokens corresponding to a specific category that is not pre-tagged, it exhibits high similarity with the dataset of the pre-rained model. In addition, a lot of words are already included in a pre-tagged form in our web service; thus, the number of datasets to be added by users is expected not to be too large. Therefore, we adopt retraining of only the classifier in this paper to achieve high performance even with small datasets that are highly similar to those used in the pretrained model.

In the example depicted in Figure 10, the term, "neurotoxic amyloid beta", is additionally tagged by the user. This term refers to the amino acid peptide of amyloid plaque and belongs to the real proteins category. However, in the original literature, it was not recognized by the pretrained model. In this situation, a dataset containing the word "neurotoxic amyloid beta" tagged in the proteins category is downloaded as a tsv file using the proposed web service. Table 3 depicts a part of a sample dataset that includes the word.

Table 3. Dataset Sample.

Word	Tag
production	O
of	O
neurotoxic	B
amyloid	I
beta	I
(Aβ)	O

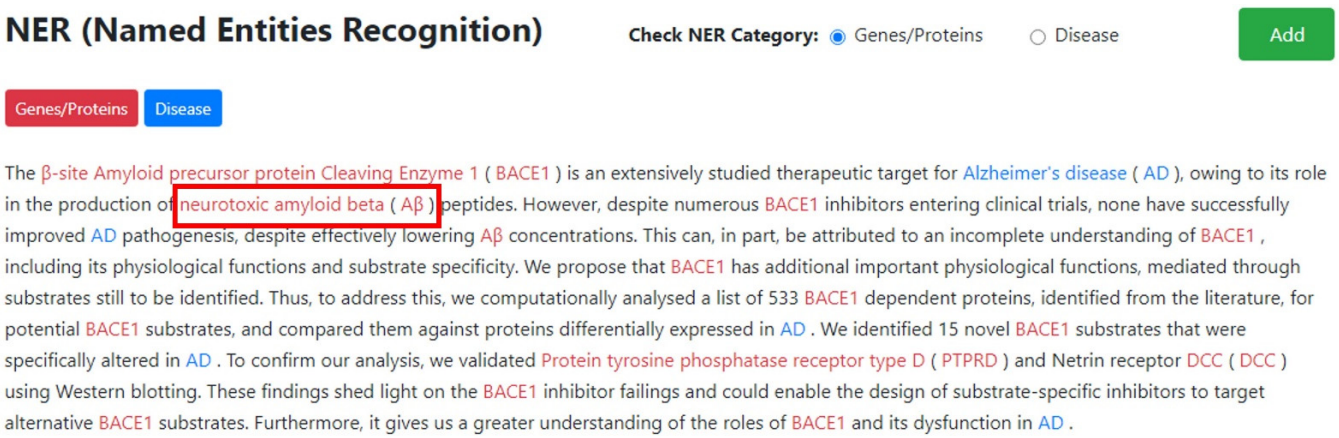

Figure 10. Web service result obtained by applying the retrained model.

We retrain the dataset using the pretrained BERT model fine-tuned for each category. The batch size is taken to be 32, the learning rate is taken to be 0.00005, and training is performed for 200 epochs in aggregate. Figure 10 illustrates the automatic tagging of the term "neurotoxic amyloid beta" by the tagging tool after retraining the model corresponding to the original PMID input.

5. Discussion

Several tagging tools have been developed to achieve high tagging efficiency by users. They have been studied particularly in the biological domain to enable collective tagging. In this paper, we propose a machine learning-based tagging tool that enables automatic tagging in specific domains, user-defined tagging, and dataset generation for fine-tuning. The proposed system yields NER and RE results for the genes/proteins and diseases categories extracted using the pretrained BioBERT model via a web service. The provided results can speed up the manual annotations of documents by both non-experts as well as experts familiar with the field of biomedical science. Users can interact with the system using the web service, enabling user-defined tagging. Further, the dataset can be fine-tuned, underlining the highly customizable performance of the model. In this way, augmenting the dataset with our tools can result in improved outcomes. Additionally, the outcomes of this system are anticipated to be applied for learning document corpora from other domains. Using our system, users can quickly build learning datasets by retrieving relevant literature and performing custom tagging to extract domain-specific terms. If they make a pretrained model by newly training it into a BERT-based model, it is anticipated that a text mining model with outstanding performance in other domains in addition to biomedical domains will be constructed. However, several problems persist. Firstly, the proposed system does not allow tagging collaboration between users. Secondly, automatic retraining and updating of the model using the created dataset is not possible. In the current system, users have to retrain the model manually by downloading the generated dataset, and then directly replacing the original model with the retrained model. Resolution of these problems is expected to yield a tagging tool system that combines the advantages of prior tagging tool systems.

6. Conclusions

In summary, a tagging tool system capable of generating datasets for fine-tuning itself is proposed in this study. It is implemented as a web service, and the effect of retraining the generated dataset is evaluated. The NER and RE results are visualized for specific domains (genes/proteins and diseases) by fine-tuning the pretrained BERT model. In addition, a novel tagging tool with a dataset generation function is proposed. The generated dataset improves user-customized recognition performance by retraining the pretrained model. However, the proposed system does not support collaborative tagging or automatic model

retraining and updating. In addition to resolving these limitations, the scope of the domain must be expanded to provide various categories of recognition results. Additionally, further research should be conducted to further improve the accuracy and speed of data curation.

Author Contributions: Conceptualization, Y.-J.P., M.-a.L. and G.-J.Y.; methodology, C.-B.S. and S.J.P.; software, Y.-J.P. and G.-J.Y.; validation, Y.-J.P., M.-a.L., G.-J.Y., S.J.P. and C.-B.S.; formal analysis, C.-B.S.; investigation, M.-a.L.; resources, C.-B.S.; data curation, Y.-J.P.; writing—original draft preparation, Y.-J.P. and M.-a.L.; writing—review and editing, Y.-J.P., G.-J.Y., M.-a.L., S.J.P. and C.S; visualization, Y.-J.P. and M.-a.L.; supervision, S.J.P. and C.-B.S.; project administration, Y.-J.P.; funding acquisition, S.J.P. and C.-B.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government Ministry of Science and ICT (MSIT) (NRF-2014M3C9A3064706).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ayush, S.; Simmons, M.; Lu, Z. Text mining genotype-phenotype relationships from biomedical literature for database curation and precision medicine. *PLoS Comput. Biol.* **2016**, *12*, e1005017.
2. Poux, S.; Arighi, C.N.; Magrane, M.; Bateman, A.; Wei, C.H.; Lu, Z.; Boutet, E.; Bye-A-Jee, H.; Famiglietti, M.L.; Roechert, B.; et al. On expert curation and scalability: UniProtKB/Swiss-Prot as a case study. *Bioinformatics* **2017**, *33*, 3454–3460. [CrossRef] [PubMed]
3. Rak, R.; Batista-Navarro, R.T.; Rowley, A.; Carter, J.; Ananiadou, S. Text-mining-assisted biocuration workflows in Argo. *Database* **2014**, *2014*, bau070. [CrossRef] [PubMed]
4. Kwon, D.; Kim, S.; Shin, S.Y.; Chatr-aryamontri, A.; Wilbur, W.J. Assisting manual literature curation for protein–protein interactions using BioQRator. *Database* **2014**, *2014*, bau067. [CrossRef] [PubMed]
5. Campos, D.; Lourenço, J.; Matos, S.; Oliveira, J.L. Egas: A collaborative and interactive document curation platform. *Database* **2014**, *2014*, bau048. [CrossRef]
6. Pafilis, E.; Buttigieg, P.L.; Ferrell, B.; Pereira, E.; Schnetzer, J.; Arvanitidis, C.; Jensen, L.J. EXTRACT: Interactive extraction of environment metadata and term suggestion for metagenomic sample annotation. *Database* **2016**, *2016*, baw005. [CrossRef]
7. Salgado, D.; Krallinger, M.; Depaule, M.; Drula, E.; Tendulkar, A.V.; Leitner, F.; Valencia, A.; Marcelle, C. MyMiner: A web application for computer-assisted biocuration and text annotation. *Bioinformatics* **2012**, *28*, 2285–2287. [CrossRef]
8. Rinaldi, F.; Clematide, S.; Marques, H.; Ellendorff, T.; Romacker, M.; Rodriguez-Esteban, R. OntoGene web services for biomedical text mining. *BMC Bioinform.* **2014**, *15*, S6. [CrossRef]
9. Wei, C.-H.; Kao, H.-Y.; Lu, Z. PubTator: A web-based text mining tool for assisting biocuration. *Nucleic Acids Res.* **2013**, *41*, W518–W522. [CrossRef]
10. Cejuela, J.M.; McQuilton, P.; Ponting, L.; Marygold, S.J.; Stefancsik, R.; Millburn, G.H.; Rost, B.; FlyBase Consortium. tagtog: Interactive and text-mining-assisted annotation of gene mentions in PLOS full-text articles. *Database* **2014**, *2014*, bau033. [CrossRef]
11. Rak, R.; Rowley, A.; Black, W.; Ananiadou, S. Argo: An integrative, interactive, text mining-based workbench supporting curation. *Database* **2012**, *2012*, bas010. [CrossRef] [PubMed]
12. López-Fernández, H.; Reboiro-Jato, M.; Glez-Peña, D.; Aparicio, F.; Gachet, D.; Buenaga, M.; Fdez-Riverola, F. BioAnnote: A software platform for annotating biomedical documents with application in medical learning environments. *Comput. Methods Programs Biomed.* **2013**, *111*, 139–147. [CrossRef] [PubMed]
13. Bontcheva, K.; Cunningham, H.; Roberts, I.; Roberts, A.; Tablan, V.; Aswani, N.; Gorrell, G. GATE Teamware: A web-based, collaborative text annotation framework. *Lang. Resour. Eval.* **2013**, *47*, 1007–1029. [CrossRef]
14. Pérez-Pérez, M.; Glez-Peña, D.; Fdez-Riverola, F.; Lourenço, A. Marky: A tool supporting annotation consistency in multi-user and iterative document annotation projects. *Comput. Methods Programs Biomed.* **2015**, *118*, 242–251. [CrossRef] [PubMed]
15. Pérez-Pérez, M.; Glez-Peña, D.; Fdez-Riverola, F.; Lourenço, A. Marky: A lightweight web tracking tool for document annotation. In *8th International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB 2014)*; Springer: Cham, Switzerland, 2014.
16. Hans-Michael, M.; Kenny, E.E.; Sternberg, P.W. Textpresso: An ontology-based information retrieval and extraction system for biological literature. *PLoS Biol.* **2004**, *2*, e309.
17. Müller, H.-M.; Van Auken, K.; Li, Y.; Sternberg, P.W. Textpresso Central: A customizable platform for searching, text mining, viewing, and curating biomedical literature. *BMC Bioinform.* **2018**, *19*, 94. [CrossRef]
18. Frédérique, S. In Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, 23–27 April 2012.

19. Ulf, L.; Hakenberg, J. What makes a gene name? Named entity recognition in the biomedical literature. *Brief. Bioinform.* **2005**, *6*, 357–369.
20. David, C.; Matos, S.; Oliveira, J.L. Biomedical named entity recognition: A survey of machine-learning tools. *Theory Appl. Adv. Text Min.* **2012**, *11*, 175–195.
21. Safaa, E.; Salim, N. Chemical named entities recognition: A review on approaches and applications. *J. Cheminformatics* **2014**, *6*, 17.
22. Islamaj, D.R.; Leaman, R.; Lu, Z. NCBI disease corpus: A resource for disease name recognition and concept normalization. *J. Biomed. Inform.* **2014**, *47*, 1–10.
23. Kwon, D.; Kim, S.; Wei, C.-H.; Leaman, R.; Lu, Z. ezTag: Tagging biomedical concepts via interactive learning. *Nucleic Acids Res.* **2018**, *46*, W523–W529. [CrossRef] [PubMed]
24. Islamaj, R.; Kwon, D.; Kim, S.; Lu, Z. TeamTat: A collaborative text annotation tool. *Nucleic Acids Res.* **2020**, *48*, W5–W11. [CrossRef] [PubMed]
25. 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. [CrossRef]
26. Wei, C.-H.; Kao, H.-Y.; Lu, Z. GNormPlus: An integrative approach for tagging genes, gene families, and protein domains. *BioMed Res. Int.* **2015**, *2015*, 918710. [CrossRef] [PubMed]
27. Wei, C.-H.; Phan, L.; Feltz, J.; Maiti, R.; Hefferon, T.; Lu, Z. tmVar 2.0: Integrating genomic variant information from literature with dbSNP and ClinVar for precision medicine. *Bioinformatics* **2018**, *34*, 80–87. [CrossRef] [PubMed]
28. Robert, L.; Lu, Z. TaggerOne: Joint named entity recognition and normalization with semi-Markov Models. *Bioinformatics* **2016**, *32*, 2839–2846.
29. Wheeler, D.L.; Barrett, T.; Benson, D.A.; Bryant, S.H.; Canese, K.; Chetvernin, V.; Church, D.M.; DiCuccio, M.; Edgar, R.; Federhen, S.; et al. Database resources of the national center for biotechnology information. *Nucleic Acids Res.* **2007**, *36* (Suppl. 1), D13–D21. [CrossRef]
30. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
31. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
32. Smith, L.; Tanabe, L.K.; Ando, R.J.N.; Kuo, C.-J.; Chung, I.-F.; Hsu, C.-N.; Lin, Y.-S.; Klinger, R.; Friedrich, C.M.; Ganchev, K.; et al. Overview of BioCreative II gene mention recognition. *Genome Biol.* **2008**, *9*, S2. [CrossRef]
33. Bravo, À.; Piñero, J.; Queralt-Rosinach, N.; Rautschka, M.; Furlong, L.I. Extraction of relations between genes and diseases from text and large-scale data analysis: Implications for translational research. *BMC Bioinform.* **2015**, *16*, 55. [CrossRef] [PubMed]
34. Sachan, D.S.; Xie, P.; Sachan, M.; Xing, E.P. Effective use of bidirectional language modeling for transfer learning in biomedical named entity recognition. In *Machine Learning for Healthcare Conference*; PMLR: Westminster, UK, 2018.
35. Yoon, W.; So, C.H.; Lee, J.; Kang, J. Collabonet: Collaboration of deep neural networks for biomedical named entity recognition. *BMC Bioinform.* **2019**, *20*, 55–65. [CrossRef] [PubMed]
36. Habibi, M.; Weber, L.; Neves, M.; Wiegandt, D.L.; Leser, U. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics* **2017**, *33*, i37–i48. [CrossRef] [PubMed]