Retrieving Knowledge of Molecular Mechanisms from PubMed Titles via an Event Extraction Approach

David Spellman[1]
Detailed Affiliations
[1]Department of Computer Science, University of Pittsburgh, Pittsburgh, PA, USA;

4/24/2023

Abstract

Natural language processing-based information extraction has the potential to have a significant impact on the field of medical research and development. Language models can be used to support the creation of question answering systems focused on molecular mechanisms. Experts in the medical field can make use of such systems to obtain meaningful answers to highly specialized questions. In this work, three main issues pertaining to information extraction from biomedical literature are addressed: 1. Molecular mechanism extraction needs to have both the structure and flexibility necessary to capture the critical features an expert would understand. 2. In the Biomedical domain, titles are the most concise summary of a paper's key findings. 3. There is a lack of annotated training data to enable a machine learning model to learn the same information seeking insights that a biomedical expert would possess.

A reasonable solution to these three points is to use an event extraction driven approach for automatic extraction of knowledge from biomedical literature, where titles are harnessed to extract molecular mechanisms. In this process, m6A literature titles were used to produce the first annotated, event extraction training data set for molecular mechanisms, composed of titles from the literature. Using this newly annotated dataset, a transformer based deep neural network model is fine-tuned using DYGIE++ (Wadden et al, 2019).

1. Introduction

Recent developments in neural language modeling such as the transformer, BERT, and GPT have made large improvements in natural language information extraction possible. Using the transformer architecture plus unsupervised training tasks such as masked language modeling (MLM), enormous unlabeled domain specific corpuses can be used to produce pretrained language models that perform well on natural language tasks falling within the specific domain they were pretrained on. This has significantly expanded the possibilities for the use of natural language processing in important fields such as medicine. Here the possibility of textual information extraction in the medical research domain is further investigated by testing to see if it can provide answers to critical mechanistic questions through the extraction of biological mechanisms from PubMed literature. One way of performing this is to use information extraction to support knowledge graph construction. In turn, knowledge graphs can allow for performing inference and information retrieval on biological mechanisms. A graph containing a concise representation of the information available in the current medical literature would be very valuable for tasks such as exploring drug to drug interactions and drug discovery. Such a knowledge graph can be produced manually. However, with the amount of new research being

produced every year it would be very time inefficient and costly to have medical experts mine the information needed. This is where natural language processing is important in order to automate the process of information extraction from text. Such techniques include named entity recognition, or the process of recognizing named objects or people and identifying what class of object or person those entities fall into. Relation extraction is then the process of taking those named entities and capturing relationships between them, such as “METTL3 is a writer of m6A”, where the relation is a triplet describing the relationship between METTL3 and m6A. These natural language techniques have been frequently attempted on entire documents or abstracts, and literature-based information retrieval able to handle extracting informationally complete biological mechanisms in a format providing the key details that an expert would look for. Current widely used tools for biological information extraction include tools such as PubTator, which can be accessed from the NIH website. PubTator will take an entire PubMed article and identifies many single named entities but provides none of the needed context for extracting biological mechanisms, and can only label entities as either gene, small molecule, species, mutation, protein, or disease. To address the lack of sufficient biological mechanism extraction techniques, this work explores a title-based approach using event extraction as the information extraction strategy of choice. Having titles as the central focus for information extraction has not been investigated enough in the medical domain, and PubMed titles can often provide a wealth of compact knowledge on biological mechanisms. Information extracted from the titles can then be harnessed to produce a knowledge graph focused on molecular mechanisms.

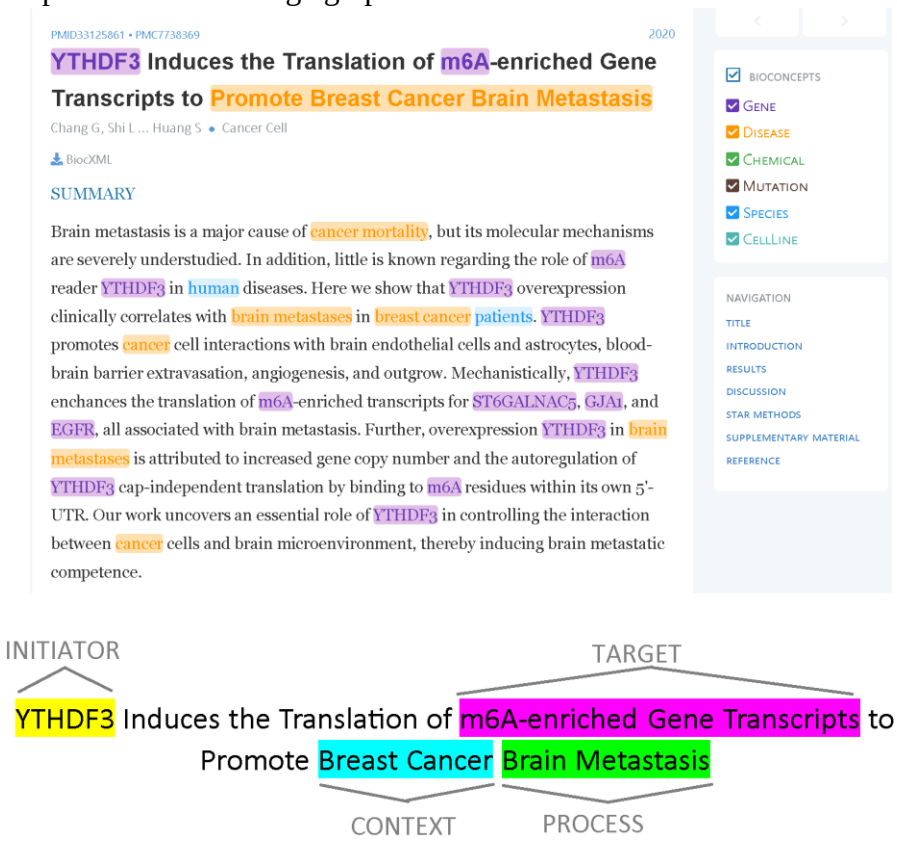


Figure 1-Top: Example PubTator named-entity-extraction, Bottom: Event extraction from title

Figure 1 illustrates how using event extraction on a PubMed title compares with using PubTator on the abstract of the same PubMed article to extract the named entities. PubTator will extract named entities from the abstract whether they are part of the key mechanism reported in the paper or not. On top of this, PubTator cannot organize these extracted entities into a mechanistic summary that incorporates the biological context of the over-all mechanism and how the key entities of the mechanism interact with each other and relate to each other. Capturing the form of regulation occurring in the mechanism is also very important for understanding the biology being extracted, and event extraction enables the capture of the important regulation verb. The second example in figure 2 shows the relation extraction of chemical-protein pairs used in (Peng et al, 2018), providing a comparison that illustrates how relation extraction from an abstract does not reliably capture the important relationships and interactions between entities in a full biological mechanism. It is flexible enough to produce an extraction template that conveys critical information in a meaningful way. The event template can accommodate any number of entities being extracted, where each entity plays a unique, abstract role in the event. The verb extracted can be used to distinguish between different events and adds additional context to a single event by expressing what is occurring between the extracted arguments.

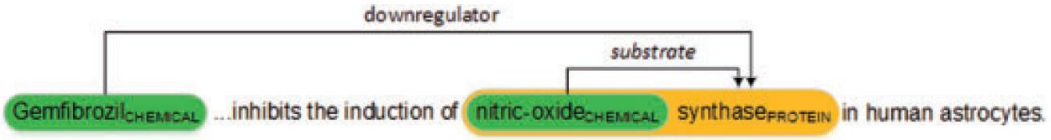
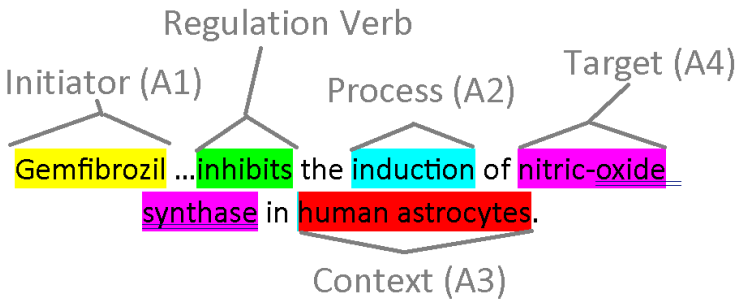


Figure 2-Sentence labelling from Peng et al, 2018



2. Related Work

Many recent examples of using machine learning for information extraction use deep neural transformers pretrained on masked language modeling and next sentence prediction tasks with huge language and domain specific datasets. These often very large, pretrained language models learn contextual linguistic patterns throughout the corpuses that they are trained on, and this information is expressed in the embeddings they produce. These latent space features expressed in the embeddings are important to predict what words or phrases are likely to occur side-by-side, what words are most similar, what word is most likely to fill in a blank, what two phrases are most likely to be contextually related, or what phrase is most likely to fall under a given

label. Some of the most popular pretrained models include the different varieties of pretrained BERT, GPT, BART, and ROBERTA models. The original transformer model included both an encoder network and decoder network, both fueled by the game changing attention layer to help improve the sequence analysis capabilities of the network over older recurrent neural network (RNN) and long short-term memory (LSTM) based techniques. However, BERT (Devlin et al, 2019) is an encoder only model that can produce these embeddings with improved bidirectional context, and GPT (Radford et al, 2018), is its mirror decoder only counterpart that can take these deep contextual embeddings and produce a textual output. For information extraction purposes encoder only transformer models such as BERT are excellent for producing the embeddings necessary for performing state of the art information extraction. There are many examples of this in recent years for named entity recognition, relation extraction, and event extraction, including (Wei et al, 2019), (Roy et al, 2021), and (Rasmy et al, 2021). Using approaches such as these that leverage the embeddings produced from language models such as BERT, significant progress has been made in areas such as medical information extraction. Nevertheless, much of the focus specific to the medical domain of information extraction has gone into clinical information extraction for tasks such as patient diagnosis assistance and patient outcome prediction. The closest medical related uses of information extraction to biological mechanism extraction are use-cases such as (Peng et al, 2018), which extracted chemical protein relations that were only part of a full biological mechanism. Furthermore, these biological mechanisms that the chemical-protein relations fall into are only one category out of many potential mechanism categories that can be found in PubMed articles. The closest subject that could be found to extraction of actual biological mechanisms was using event extraction to gather COVID mechanisms from paper abstracts (Hope et al, 2020). However, this work focused on mechanisms in more of a general sense instead of strictly a biological sense. Some of the mechanisms extracted were biological but did not capture the important details that would be desired for a biomedical expert to answer questions about complex mechanistic interactions between drugs, RNAs, proteins, genes, pathways, and diseases.

Some newer approaches for event extraction includes works such as (Liu et al, 2019), (Wadden et al, 2019), and (Wang et al, 2021). These works all try to leverage the developments in large, pretrained, neural, language models to greatly improve the accuracy and flexibility of event extraction over the state-of-the-art approaches that existed before developments such as BERT and ELMO (Peters et al, 2018). These papers take the embeddings produced from language models such as BERT and ELMO and try to improve event extraction by enhancing the latent embedding space through additional contrastive pretraining, (Wang et al, 2021), using such models for zero shot transfer learning to increase the scope of what events can be reliably extracted without having to produce additional expensive annotations, (Huang et al, 2018), or by adding additional algorithms to help boost event extraction with language models (Wadden et al, 2019). When it comes to domain specific information extraction, enhancement of the embedding space via use of extensive unlabeled domain specific textual examples is important, since this enables the embeddings to capture as many critical informational features of the subject in question as possible. Contrastive learning in particular has been shown to be an effective way of doing this on top of the pre-training tasks such as masked language modeling that are already standard, and papers such as (Wang et al, 2021) show the additional improvement that language

models can gain. Contrastive learning helps by further separating features that represent stark contextual and conceptual differences in the latent space, and then further congregating features that should be closely associated together in latent space. For example, in (Wang et al, 2021), the purpose is to use the sentence context included in word embeddings in order to help pull correct verb-entity pairs that belong in the same event-mention closer together in the latent embedding space, while further separating verb-entity mismatches that are a part of different event-mentions. The contrastive learning described here will help a pretrained language model avoid extracting incorrect verb-entity pairs when performing event extraction, and this type of learning could also be harnessed to help improve other information extraction tasks such as relation extraction. DYGIE++, a framework introduced in (Wadden et al, 2019), performs several common information extraction tasks including named entity recognition, relation extraction, coreference resolution, and event extraction. The technique produces deep contextual embeddings with BERT and performs well on both sentence as well as cross-sentence level information extraction tasks. *DYGIE++* extends a previous DYGIE paper (Wadden et al, 2019), where a dynamic span graph algorithm was introduced to help solve coreference issues as well as to help mine out entity relations. Coreferences are clusters of words that all refer to the same entity, or groupings of event mentions that all refer to the same event conceptually, and up until now coreference resolution is still a challenging issue in natural language information extraction. Resolving coreferences is likely to be unneeded when extracting information title-by-title but would be very useful when extracting information from abstracts, entire documents, or even groups of documents. In DYGIE++ this dynamic graph algorithm was improved to accommodate event extraction by generalizing the algorithm to cover span pairs focused around trigger verbs and their related entities. The dynamic span graph algorithm functions greedily, by using the embeddings produced by BERT to produce all potential entity and verb pairs in a sentence or series of sentences, and starts by producing a fully connected span graph. Next, the algorithm goes through a pruning phase, where spans in the graph that are predicted to have a low probability of being part of the event being extracted are removed. For event extraction, the algorithm trains a fully connected neural pruning network for each argument type being extracted and optimizes a set of thresholds in order to determine which spans are removed from the graph and which remain. The DYGIE++ framework fine-tunes BERT while also training the neural classifiers for the pruning step of the dynamic span graph algorithm. The greedy nature of this algorithm could prove inefficient for extraction tasks dealing with large bodies of text, but with a title level extraction task there is the potential that this span graph algorithm could perform very well on tasks such as title level event extraction. BERT performs bi-directional sequence detection on sentences at the word level, providing context on sentence structure, word usage context, and word meaning context in both the forward and backward direction. This learning of deep context through improvement of word embeddings is how many current language models capture knowledge of and construct a hypothetical understanding of subjects and concepts in the field of computational linguistics. This helps the model learn word context, and to help generalize the word encodings in order to make classifying words that are rarely seen more accurate. Obtaining better performance on infrequently encountered tokens is necessary in the biomolecular domain since there are many entities that are important to mechanisms that

appear with low frequency in the wider biological corpus. There will likely always be new genes, RNAs, small molecules, and pathways being reported in newly published research.

DYGIE++ was used for a purpose similar to event extraction of biological mechanisms, but not from titles in (Hope et al, 2020). This work uses abstract based information extraction to build up a diverse knowledge base of mechanisms related to Covid-19 whether molecular or economic. Since this source also uses DYGIE++ to fine-tune BERT and train an event extraction classifier, it can serve as a good measuring stick for determining the merits of the alternative title level extraction when it comes to accuracy. If the title-based approach exceeds the eighty percent accuracy reported by (Hope et al., 2020), then there is evidence for title-based approaches being promising merely due to the fact that a higher precision and recall can be achieved. Looking at this example of Covid-19 knowledge extraction can also be helpful as a guide for considering how to break down our single regulation event into sub-events, or how we could expand our model to use the abstract of articles for supplementing our title-based knowledge extraction. To consider what could then be performed with a knowledge graph produced with molecular mechanisms extracted from literature titles (Li et al., 2020), discusses many recent algorithmic uses. These uses could span from performing network propagation to feature engineering for network-based machine learning. Information retrieval only scratches the surface of what could be done with a knowledge graph of molecular mechanisms; with more recent developments making graph representation learning a possibility. These knowledge graph-based techniques can all aid in investigating disease, new drug possibilities, and improving precision medicine.

Methods

In order to define an event extraction template for biological mechanisms, a set of regulation verbs were identified that signified some form of molecular mechanism. Using these regulation verbs, a single flexible event template was designed in order to capture the diverse entities that can be involved in biomolecular events. Since there is currently only a single template and single event type, the regulation verb does not serve as a trigger as often seen in standard ACE event extraction, but it is still important to extract the regulation verb for the purpose of knowing what form of regulation is occurring in the mechanism. This flexible template includes arguments that can be filled by any of the following: genes, cellular and molecular processes, cell types, diseases, proteins, RNAs, chemicals, pathways, physiological conditions, specie names, and organ types. There are four of these critical arguments, and all arguments have some critical relationship to both the regulation verb and other arguments as defined by the biological mechanisms event template. Argument one is best described as the initiator, which is most often a gene, but can also be a protein, RNA, small molecule, physiological condition, or in some cases a sub-event that must occur in order to cause the mechanism to become active. The initiator is the argument that causes the mechanism to occur and can be viewed as the upstream argument that causes the mechanism. Argument two is the process, or the critical entity that the mechanism is regulating. The process is the entity being modified in the title by the regulation verb and is ultimately the entity being affected by the mechanism. The third argument is the context, best described as the argument that adds additional critical information to the process, such as what disease the process occurs in, what organ is affected by the process, or more

information about the identity of the process. The context argument is usually a disease, but can also be an organ, species name, RNA, or protein, depending on the scope of the biological mechanism, and whether it is describing something on more of a micro or macro scale biologically. The fourth and final argument is the target, or the entity that is acted upon by the initiator argument in order to cause the mechanism. The target argument is most commonly a gene, but it can also be an RNA, protein, pathway, or more specific site on some protein or RNA. Using these definitions, the full event template can then be described as; the initiator acts upon the target to regulate the process in the provided context.

In order to fine-tune BERT and train neural classifiers for the different parts of the biological mechanism event template, the M6a dataset was used. This dataset includes 500, research, article titles scraped from Pub Med that contained certain key words such as m6A related gene names and protein names. The first draft of the event template included the regulation verb plus five arguments, where one of these arguments was the core molecular mechanism underlying the broader biological mechanism. However, it was decided to remove this mechanism argument from the template since the target and mechanism arguments can often be difficult even for a human annotator to distinguish. Plus, the mechanism argument can be extracted as a second step once the rest of the arguments have already been extracted; by treating the mechanism entity as a relation of the target entity. A simple example can illustrate how the original five argument template worked, and how the new four argument template compares. For instance, a title from the dataset states, “METTL3 Promotes Esophageal Squamous Cell Carcinoma Metastasis Through Enhancing GLS2 Expression.” In this example argument one or the initiator gene is “METTL3”, the regulation verb is “Promotes”, the argument two or process is “Metastasis”, the argument three, disease name, or context is “Esophageal Squamous Cell Carcinoma”, the argument five or mechanism is “Expression”, and the argument four or target gene is “GLS2.” In the updated template argument five or the mechanism is simply removed, and a second post processing extracting step is used instead to obtain the additional entity “expression”. The mechanism is often some form of molecular signaling, stabilization, degradation, modification, or other biochemical event acting upon the target, such as the expression of the target. Sometimes the mechanism can also be a downstream product of the initiator argument acting upon the target, such as a signaling pathway that becomes active in order to cause the form of regulation described by the regulation verb to occur. In some less common instances the process can be a process on the molecular level such as stabilization, which in that case the location will be a protein or RNA molecule instead of a disease or cell type.

It is common that in these titles not all arguments are present, and in some cases this means that argument classification can be a bit ambiguous, but the annotations were performed so that the arguments were always occupied by entities that would produce a permissible knowledge representation if reviewed by a biomedical expert. It should also be noted that in the future information extraction from the abstracts will likely be added to this pipeline to improve the process of mechanism extraction by filling in any holes present in the title. Several passes of annotation were performed in order to refine the dataset, and to make the annotation consistent across all cases where the event template could be filled in multiple valid ways. For example, this title possesses enough ambiguity that the template could be filled two different ways,

“Enterotoxigenic Escherichia coli infection promotes enteric defensin expression via FOXO6-METTL3-m6A-GPR161 signaling axis.” Here “Enterotoxigenic Escherichia coli infection” is clearly the initiating condition that fits in argument 1, “promotes” must be the regulation related verb serving as the trigger, “enteric defensin expression” is the process that is being regulated, but “FOXO6-METTL3-m6A-GPR161 signaling axis” could be the mechanism (argument five), or the target (argument four) of some other mechanism. Conflicts such as this one were part of the reason for removing argument five, and only extracting the first four arguments during the event extraction phase of the overall biological mechanism extraction pipeline. The current single event template also cannot perfectly accommodate some complex titles. Commonly this is because of four issues. 1. The initiator gene or molecule is tied to some more conceptually complex condition that allows it to facilitate the mechanism described in the title. For instance, “Downregulation of m6A Reader YTHDC2 Promotes the Proliferation and Migration of Malignant Lung Cells via CYLD/NF-kappaB Pathway.” In this title “YTHDC2” does not directly promote “Proliferation and Migration”, and the fact that the gene being downregulated is what initiates the mechanism is a detail that is mandatory to capture in order to correctly extract the knowledge contained. For now the condition is simply extracted as the span for the initiator argument, but moving forward a second step should be designed to help decompose these cases with complex initiators. At one point having an argument zero in the template was even considered in order to capture entities such as the term “degradation” in this particular example, but this idea was set aside for reasons similar to those that caused argument five to be removed from the template. These complex initiators will be handled later in the pipeline when other entities such as the mechanism, (argument five), are delt with as a post-processing step. 2. There are a significant number of titles that contain two regulation mechanisms, or a mechanism that is involved in the function of a larger mechanism as a sub-mechanism. In many of these more complex multiple mechanism titles the multiple mechanisms come in the form of an extra target and mechanism to add more indirection to the molecular mechanism depicted. 3. A small number of titles do not fit the template because the mechanism described cannot be considered in terms of biomolecular regulation or expresses some more nuanced pattern of regulation that cannot be expressed by filling in the current template that expresses only a single one-way form of regulation. An example of this are some rare titles where a mechanism with a cyclic form of complementary regulation is described where one gene up-regulates a process while another down-regulates the same process. For now, the template can only extract a single initiator and a single regulation verb.

Initial training was performed with DYGIE++, (Wadden et al, 2019), as described in the related work section. This initial round of experiments included only fine-tuning of BERT, and training of the necessary event extraction classifiers as used in the DYGIE++ framework. For this training all models were trained with 396 of the total 500 titles in the M6A dataset, including all of the titles that contained at least one regulation verb along with two or more of the four arguments extracted by the revised event template. All models were trained using 5-fold cross-validation, where three of the five splits were always combined for the training set, one split was used as the development set, and the final split was held out as the test set for that particular fold. All test metrics for each model were averaged across the 5 folds, so that all models were evaluated on the entire dataset when combining evaluation across all five folds. Three different

baseline model types were trained; one using PubMed BERT Base Uncased as the encoder, the second using Clinical PubMed BERT Base Uncased as the encoder, and the third using PubMed BERT Large Uncased as the encoder. The combined predictions on the full dataset across all five folds were scored as true positives, false negatives, and false positives using the standards outlined by the state-of-the-art ACE event extraction literature, (ACE 2005). This means that each argument and the regulation verb are treated as their own class in a multi-class classification one verses rest manner. True positives are counted as predictions where the span indices and the class of a prediction matches the span indices and the class of a ground truth label. If a ground truth label for a given instance has no prediction that matches both its class and span indices, then that ground truth label is counted as a false negative. Otherwise, any prediction that does not perfectly match a ground truth label is counted as a false positive. The true positives, false positives, and false negatives were then used to calculate precision, recall, and F1 metric for all arguments and the regulation verb across all three models. ROC curves were produced for the regulation verb and each argument using the ROC sklearn implementation, and the AUROC was calculated using the same package.

In order to show how the M6A dataset was affecting the performance of the models; further experiments were run using varied percentages of the available training data. This meant using 5-fold cross-validation as in the first batch of training experiments; however, this time random shuffling of the dataset was not used in order to ensure that all models saw the same training examples across all five folds for the same percentage of training data used. For example, when only ten percent of the training data was used for an experiment, this meant that the approximate 240 training examples available during each of the five folds of cross validation was instead restricted to twenty-three. The experiments were structured so that all three model types saw the same twenty-three training examples for each fold one through five, so that random reshuffling could not give one model a random advantage over the others based on which training samples it got over the five folds. This same principle was maintained as more training samples were added for higher percentages of training data. All three models were trained with ten, twenty, forty, seventy, and one hundred percent of available training data. The test and development splits remained the same as in the previous experiments, except for the fact that they would contain the same samples through folds one to five since the random shuffling of the dataset had been removed for these experiments. All the same metrics were calculated for these train data variation experiments as were calculated in the first round of baseline training experiments, and the same methods for calculation were used.

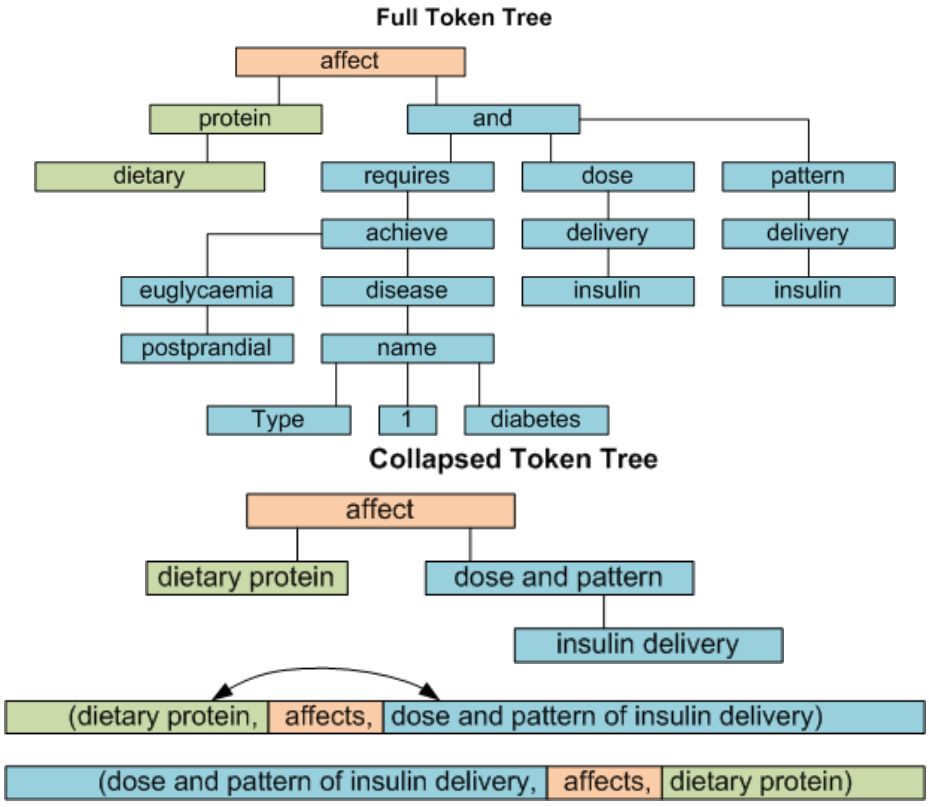


Figure 3- Figure showing example of contrastive learning negative pair.

The results from CLEVE (Wang et al, 2021), inspired additional experimentation to see if contrastive learning could be harnessed to obtain improvement in extraction precision, recall, F1, and AUROC over the baseline models that use three different pretrained BERT encoders. The idea came from the fact that CLEVE used AMR parsing in order to perform trigger-argument pair swaps to produce negatively augmented contrastive examples. These negatively augmented contrastive examples were then used to perform contrastive pretraining to help improve correct trigger argument pair prediction in event extraction. The idea behind CLEVE was to help pull correct trigger-argument pairs together in latent space, while further separating incorrect trigger-argument pairs. However, the Biological mechanism extraction does not use trigger classification since there is only a single event template, but the concept of using AMR parsing was adopted in order to produce negative pairs. The process of producing these negatively augmented titles is shown in the figure above. Since extraction is being performed only on the title, it is much easier to leverage the sentence syntax structure to reliably swap some important entity spans around the regulation verb, and these entity spans are very likely to be arguments in the biological mechanism event template. The AMR parse tree is processed by first identifying the regulation verb closest to the root of the tree, and then identifying the two entity spans that branch off immediately on either side of the verb. As shown in the figure, these entity spans can be traversed and joined together into single nodes on either side of the verb, and then swapped with one another in order to form a negative augmentation of the title that is informationally incorrect. The hypothesis behind doing this was to see if BERT could better learn the patterns behind the arguments in the event template by training on distinguishing informationally correct titles from

titles that have become nonsense because of the swap between entities on either side of the regulation verb. Positive augmentations were then made by using word-net and bio-verb-net to replace words in the title with one of their closest synonyms in order to make simple augmentations of the original titles where the informational value of the title was unchanged. To further help with the contrastive learning, the supervised version of the code from Sim-CSE was adopted in order to hopefully help boost performance, and to help supplement the positive examples with their in-batch drop-out driven data augmentation. They perform this by producing embeddings for examples twice during each batch of training, embedding a data instance once, performing some amount of minimal, random dropout on the network, and then embedding the same example a second time post drop-out. The network is reverted to post drop-out after the embeddings for an example are produced, and the result of this minimal random drop-out is a small amount of augmentation that equates to positive augmentation. The supervised setting of Sim-CSE was run using eighty-five thousand unannotated PubMed titles, their augmented hard-negative equivalent produced with the AMR parser, and the augmented positive example produced by the synonym substitution on all three pretrained BERT models used in the previous two experiment types. Once each BERT model was contrastively pretrained on this task, the first experiment type with all available M6A training examples was re-run with the contrastively pretrained BERT models. The results from this third experiment were evaluated using the same metrics as the other two experiments, and all results from contrastive pretraining across the three BERT model types were compared with the three baselines.

Results

The results from the second training variation experiment show that across the three pretrained BERT types that precision, recall, and F1 often rapidly increases with more training examples up until using about ten to twenty percent of available training examples during 5-fold cross-validation, translating into about twenty to fifty training title examples. It can be seen that for most arguments there is an inflection point in the range of twenty to fifty titles used where performance gains from adding additional titles after this point are far less and would likely only be significant if hundreds or even thousands more training titles were annotated and available for training. On the other hand, experiments were run using only a single training title per fold, and when this was performed all three models could not achieve a precision, recall, or F1 greater than or equal to 0.01. When looking at the AUROC plots, adding additional titles for the most part had no clear positive or negative trend. The AUROC fluctuated some between ten and one-hundred percent training, but there were no other interesting details unless arguments or the regulation verb are examined in isolation. When looking at specific arguments or the regulation verb in isolation, there are some more interesting trends of greater magnitude. For example, some arguments gained a lot in precision or recall by adding more examples for a particular BERT model, but then the same argument when using a different BERT model did not see the same boost in performance with more training titles. Also, for some models the regulation verb also saw a significant drop in classification performance when more training examples were used. In some cases arguments or the regulation verb saw a big drop in performance around fifty percent, and then the performance climbed back up to a maximum performance at one-hundred

percent available training examples. Likely reasons for these differences include the fact that these BERT models have seen different pretraining data from slightly different corpuses, and one BERT model might have a better pretrained latent understanding of a given argument type to begin with over another BERT model. Some training examples in the dataset might be more confusing or less well annotated than others, so if a BERT model already has a good latent representation for a certain feature type, less informative training examples might instead serve to confuse BERT or slightly damage the latent space representation until BERT sees more examples that allow it to climb out of a local maximum with respect to error. However, over all the additional training examples have no real negative effect on AUROC, and serve to boost precision, recall, and F1 across the board. This shows that well labeled event extraction datasets such as the PubMed title M6a datasets are necessary, and that current pretrained language models cannot perform event extraction as well in a zero-shot setting.

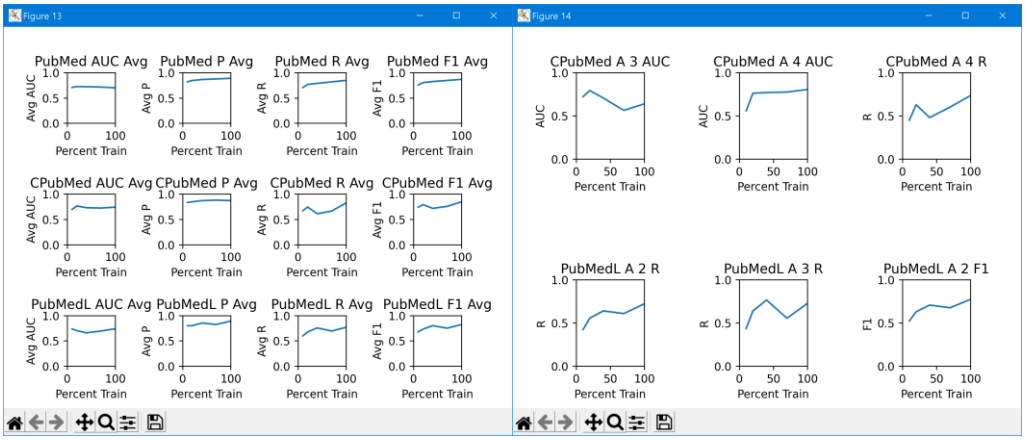


Figure 4-Graphs comparing performance of pre-trained model.

The results from using contrastive learning as an extra pretraining task compared with using no contrastive learning are shown in the tables above. For the PubMed BERT Base Uncased model, the contrastive learning had a negative effect on the performance of the model when utilized for producing embeddings during event extraction. Most metrics across the arguments and regulation verb fell by a noticeable amount. However, the PubMed BERT model performs the best out of the three baseline models. The other model with only twelve attention layers, Clinical PubMed BERT Base Uncased, did see a sizable improvement through contrastive learning on precision and F1, but the recall saw no large change, and the AUROC went down across most arguments and the regulation verb when contrastive learning was added. The best overall model was PubMed BERT Large, with twenty-four attention layers, when contrastive learning was used to pretrain it. With contrastive learning saw an improvement for PubMed BERT Large across precision, recall, and F1 by sizable amounts; and even the AUROC saw improvement on some arguments. The only downside is that the AUROC's for PubMed BERT Large with contrastive learning are not better than the Clinical PubMed BERT model without contrastive learning. However, one explanation for these low AUROC results could be the punishing nature of how true positives and false positives are determined for ACE event extraction evaluation, and the fact that a prediction span that is mostly correct is labeled a false

positive if only a single token from the label is excluded, or if a single token that was not in the label is included. This means that arbitrary tokens that do not really affect the correctness of the extraction can come into play and cause essentially correct predictions with high output probabilities to be scored as false positives. These rare instances affect the AUROC score more dramatically than the precision, F1, and recall, since it prevents a reasonable threshold from dividing a greater majority of the true positives from the false positives. However, one way of correcting these issues would be to make the labels in the M6A dataset more minimalistic, capturing only the critical tokens for event extraction to be informationally complete. This potential improvement of the M6A dataset will be discussed further in the future work section. A good explanation for why PubMed BERT Large saw a very significant boost in performance from contrastive learning while the other two BERT models did not, could be the fact that PubMed BERT Large has twice as many attention layers, and thus may be able to more easily improve its embedding space without hitting local sub-optimal embedding representations that it cannot optimize out of. However, it could also be that the data this model was trained on gave it the contextual knowledge to come out of the contrastive pretraining without having its embedding space confused and was instead able to gain more contextual knowledge of the specific entity features involved in the biological mechanism event extraction.

Model	**Trigger AUROC**	**Arg1 AUROC**	**Arg2 AUROC**	**Arg3 AUROC**	**Arg4 AUROC**	**Status**
PMB with CL	0.698	0.756	0.571	0.595	0.696	worse
PMB no CL	0.697	0.810	0.636	0.659	0.690	Better
PMBL with CL	0.699	0.741	0.656	0.740	0.866	Argument trade-off
PMBL no CL	0.617	0.846	0.745	0.719	0.760	Argument trade-off
CPMB with CL	0.623	0.718	0.628	0.657	0.691	worse
CPMB no CL	0.762	0.827	0.675	0.638	0.804	Better, best over all model

Figure 5-AUROC comparison of pretrained models with and without contrastive learning.

Model	**Trigger Precision**	**Arg1 Precision**	**Arg2 Precision**	**Arg3 Precision**	**Arg4 Precision**	**Status**
PMB with CL	0.977	0.879	0.885	0.899	0.798	worse
PMB no CL	0.977	0.912	0.852	0.881	0.818	better
PMBL with CL	0.990	0.965	0.957	0.937	0.927	Much better, best over all model by far
PMBL no CL	0.976	0.898	0.834	0.853	0.890	Worse than CL counter-

						part, but still performs well compared to others
CPMB with CL	0.971	0.903	0.867	0.868	0.812	Better, one of the better models
CPMB no CL	0.969	0.894	0.851	0.862	0.775	worse

Figure 6-Precision comparison of pretrained models with and without contrastive learning.

Model	**Trigger Recall**	**Arg1 Recall**	**Arg2 Recall**	**Arg3 Recall**	**Arg4 Recall**	**Status**
PMB with CL	0.962	0.855	0.822	0.851	0.753	A little better
PMB no CL	0.967	0.887	0.794	0.828	0.753	A little worse
PMBL with CL	0.985	0.954	0.924	0.927	0.867	Much better, best over all by far
PMBL no CL	0.946	0.826	0.721	0.726	0.616	Much worse
CPMB with CL	0.954	0.874	0.799	0.805	0.688	Argument trade-off
CPMB no CL	0.954	0.861	0.762	0.805	0.734	Argument trade-off

Figure 7-Recall comparison of pretrained models with and without contrastive learning.

Model	**Trigger F1**	**Arg1 F1**	**Arg2 F1**	**Arg3 F1**	**Arg4 F1**	**Status**
PMB with CL	0.969	0.867	0.852	0.875	0.775	Worse
PMB no CL	0.972	0.899	0.821	0.854	0.784	Better
PMBL with CL	0.987	0.960	0.940	0.932	0.896	Much better, best over all model
PMBL no CL	0.961	0.860	0.773	0.784	0.728	Much worse
CPMB with CL	0.963	0.888	0.831	0.836	0.745	Very slightly better
CPMB no CL	0.961	0.877	0.804	0.833	0.754	Very slightly worse

Figure 8-F1 score comparison of pretrained models with and without contrastive learning.

When taking the best model overall, the model using PubMed BERT Large with contrastive pretraining as the encoder to produce event extraction embeddings, the overall metrical comparisons for this technique are very promising when compared with the baseline results when using DYGIE++ provided in (Wadden et al, 2019). The F1, precision, and recall

were very good for the regulation verb, which equates to the performance on trigger identification in the original DYGIE++ reported results. The arguments also have a high F1, precision and recall, when compared to the previous reported results on argument classification for DYGIE++. Compared to the event extraction results from (Wadden et al., 2019) our reported F1 for argument classification is 0.93 a healthy improvement over their reported F1 of 0.51 on argument classification. Here we can see that extracting from single sentences with far less variation while using a single event template is more reliable for high precision and recall.

3. Conclusion and Future Work

This title based biological mechanism event extraction technique shows much promise for future information retrieval and inference systems, but there is still a lot that can be explored and refined when it comes to this on-going research. Although much time and care went towards producing quality labels for training and evaluation, the labels are still far from perfect. This is where the model can sometimes help identify inconsistencies in the annotations when reviewing the summary of the differences in the actual and predicted filled templates. For example, if the labeler missed labeling a process or outcome such as “apoptosis” in the actual outcome argument value, the model might predict “apoptosis” as the outcome argument anyway. This makes reviewing the results of the model thoroughly important, also because the extraction summary can help identify title structures that the model struggles with. As briefly mentioned previously, it may also be very important to simplify the gold labels in the M6a dataset, so that the labels only include tokens that are absolutely cruisial to the informational completeness of the event extraction. Otherwise, the annotations may have labeling inconsistancies involving tokens that are not informationally critical, that could cause unnecessary roadblocks for the model during learning. On top of this, having unnecessarily wordy annotations will make it more likely the model is punished during evaluation for not including or including tokens that have no real baring on whether extraction is correct from a biological information prospective. There may also be room for improvement by designing sub-templates of regulation instead of limiting extraction to a single template. This could help if using multiple templates offers a clearer dividing line between arguments.

One observation related to the annotated dataset that could be important is that an argument such as the initiator (argument 1), is present at a higher rate in the annotated dataset than the mechanism (argument 4). There are no titles describing a regulation event that are without an argument 1, but there are plenty of titles that capture a regulation event without providing argument 4. This means that data imbalance could potentially be hindering the evaluation or performance of the model. This issue is common in NLP, especially since the classification often contains far more negative samples than positive samples (Munkhdalai et al, 2015). However, the only real way that this class imbalance could be tackled would be to trim the M6a dataset down to titles that only contain all four arguments, which would majorly decrease the size of the dataset. However, moving forward the best answer to this possible issue would be to try to improve on the contrastive semi-supervised learning, so that the model can benefit from exposure to more augmented positive examples of all arguments in the larger unlabeled dataset, as well as more augmented informationally incorrect hard-negative examples.

A deeper literature search could be undertaken to try to find a different model architecture that proves more ideal for our use case, whether this means using a different transformer based model for embedding, different classifier for extraction, or an algorithm other than the dynamic span graph technique for event extraction.

Title level knowledge extraction could work well as the ignition step for textual information extraction in the future. The limited variation in title structure and the concise nature of the title can function as an information extraction guide if it does not contain the critical knowledge of a Bio Medical document outright. If the important information to be identified about a molecular mechanism is not in the title, then the abstract can be harnessed as a next step to help fill in gaps. In order to make use of the minority of titles that do not fit the current event template, either an improved expanded template can be designed, or additional templates can be added to expand the number of biological mechanisms that the pipeline can extract. More ablation studies will be conducted to further improve the initial model. Once the pipeline can reliably extract knowledge of molecular mechanisms, then investigating ways to construct a knowledge graph can be moved onto. With a knowledge graph that correctly represents knowledge of molecular mechanisms we can experiment with search algorithms and potentially even graph learning. Our final goal is to assess what level of information retrieval and inference can be performed with our strategy. Furthermore, we plan to assess whether there exists a transformer-based decoder such as a PubMed GPT that could be fine-tuned to directly answer research questions about biological mechanisms. This is important in order to obtain an idea about whether it is yet possible to achieve successful information retrieval and inference without the knowledge graph via some form of a Chat GPT with knowledge domain specific to biological mechanisms. However, in order for this to be possible some form of a pretrained decoder needs to be able to produce well summarized answers to mechanistic questions without producing answers that are untrue but sound like they could be real. However, whether the final solution to an information retrieval and question answering system for medical research lies in a knowledge graph or transformer decoder-based system, there could be important ethical related questions when it comes to the possibility of incorrect or misleading answers. Even if such mistakes were rare in theory, it would still be important for the system to be able to provide some form of citation for its answers, such as the paper or papers that the system extracted its answers from. This way researchers could easily sanity check results if they seem at all questionable, by checking the sources of origin provided. With the knowledge graph solution this could be easily implemented, by tagging regions of the knowledge graph with the PubMed articles they originated from. Each answer to a query could then include top PubMed articles of origin with the resulting answer, prioritizing the PubMed result list by how recent a publication is, and how much a given publication has been cited.

Starting from the title of a paper is an improvement, since annotation is far less tedious, biomolecular Pub Med titles are informationally rich, and a high accuracy of information extraction is much more easily achieved. Cross references are rarely an issue of concern within an individual title and can only become an issue when expanding extraction to include the abstract. Despite the fact that in cases details of the molecular mechanisms are left out of titles, the information extracted from the titles can be used as an information seeking index into the

abstract, to help guide further information extraction if a complete mechanism is not present in the title.

4. References

- ACE (Automatic Content Extraction) English Annotation Guidelines for Events, Version 5.4.3 2005.07.01, Linguistic Data Consortium. http://www.ldc.upenn.edu/Projects/ACE/
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Tom Hope, Aida Amini, David Wadden, Madeleine van Zuylen, Sravanthi Parasa, Eric Horvitz, Daniel Weld, Roy Schwartz, Hannaneh Hajishirzi. Extracting a Knowledge Base of Mechanisms from COVID-19 Papers. arXiv (https://arxiv.org/abs/2010.03824), 2020
- Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare Voss. 2018. Zero-Shot Transfer Learning for Event Extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2170, Melbourne, Australia. Association for Computational Linguistics.
- Michelle M. Li, Kexin Huang, Marinka Zitnik. Graph Representation Learning in Biomedicine. arXiv, 2021.
- Xiao Liu, Heyan Huang, and Yue Zhang. 2019. Open Domain Event Extraction Using Neural Latent Variable Models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2860–2871, Florence, Italy. Association for Computational Linguistics
- Munkhdalai, T., Li, M., Batsuren, K. *et al.* Incorporating domain knowledge in chemical and biomedical named entity recognition with word representations. *J Cheminform* **7** (Suppl 1), S9 (2015). https://doi.org/10.1186/1758-2946-7-S1-S9.
- Yifan Peng, Anthony Rios, Ramakanth Kavuluru, Zhiyong Lu, Extracting chemical–protein relations with ensembles of SVM and deep learning models, *Database*, Volume 2018, 2018, bay073, https://doi.org/10.1093/database/bay073.
- Peters, Matthew, et al. "Deep Contextualized Word Representations." Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018. Crossref, https://doi.org/10.18653/v1/n18-1202.
- Radford, Alec and Karthik Narasimhan. "Improving Language Understanding by Generative Pre-Training." (2018).

- Rasmy, L., Xiang, Y., Xie, Z. *et al.* Med-BERT: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *npj Digit. Med.* **4**, 86 (2021). https://doi.org/10.1038/s41746-021-00455-y
- Roy, Arya. Recent Trends in Named Entity Recognition. 2021. https://arxiv.org/abs/2101.11420
- David Wadden, Ulme Wennberg, Yi Luan, Hannaneh Hajishirzi. Entity, Relation, and Event Extraction with Contextualized Span Representations. arXiv (https://arxiv.org/abs/1909.03546), 2019.
- Ziqi Wang, Xiaozhi Wang, Xu Han, Yankai Lin, Lei Hou, Zhiyuan Liu, Peng Li, Juanzi Li, and Jie Zhou. 2021. CLEVE: Contrastive Pre-training for Event Extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6283–6297, Online. Association for Computational Linguistics.
- H. Wei et al., "Named Entity Recognition From Biomedical Texts Using a Fusion Attention-Based BiLSTM-CRF," in IEEE Access, vol. 7, pp. 73627-73636, 2019, doi: 10.1109/ACCESS.2019.2920734.