An Ensemble of Neural Networks to Identify Drug Names Linked to a Target

Kailing Wang[12] (kailing.wang@nyu.edu)

Instructors: Yoann Mamy Randriamihaja[1], Bill McMahon[1]

[1] *Computational Research, Roivant Sciences Inc.*

[2] *Center for Data Science, New York University*

Abstract

The most common mechanism of action for drugs is modifying the activity of a protein that is encoded in the human genome. Unfortunately, there is no exhaustive summary of the scientific data surrounding all molecules that have demonstrated modest or strong activity of that type. A starting point for such a summary is identifying the names of all molecules mentioned in scientific literature that have such demonstrated activity.

In order to expand our current entity library, we developed a Named Entity Recognition (NER) system trained on PubMed abstracts to extract associated drug-target pairs. By applying an ensemble neural networks approach, we not only manage to exclude ineligible entities, but also largely enhance the recall.

1 Introduction

This capstone project is to establish a document analysis pipeline where given a list of targets of interest, we improve our NER system to find new drug names acting on those targets from millions of medical papers.

The company has made efforts on this problem previously, that is building a deep neural networks NER purely based on the syntax of documents. Given a sentence containing the target of interest, the pure NER will output all the words that are classified as drug names. However, there exists a potential flaw in this method -it fails to justify the relationships between the target and drug names detected. Chances are that some drug names, which happen to appear in the sentence, are not necessarily relevant with the given target. Instead of simply obtaining a great quantity of drug names, we are aimed at developing a drug name dictionary that links each drug to a specific target that it acts on.

To solve this issue, we leverage *ChEMBL* [1] database, which is a manually curated medical database. As ChEMBL contains thousands of drug-target pairs which are proved to be related from a given publication reference, we could take advantage of the sentence structures from those reference documents, and thereby expect that this external assistance from ChEMBL could expand our drug NER capabilities.

The methodology and outcome of this project could easily be generalized into other scenarios -provided a list of entities (targets), making recommendations of entities pairs (drug-target pairs) from unstructured data.

[1] https://www.ebi.ac.uk/chembldb/

2 Data Preprocessing

From ChEMBL's list of documents, we extract two types of sentences: "Good sentence": containing both the corresponding drug and target; "Other sentence": containing the target only without any drug. We preprocessed 3011 sentences in total, with 1599 "Good" and 1412 "Other". We use the standard train-test split: 56% training set, 14% development set and 30% testing set.

For each sample, we have three types of labels: (1) sequential label marking the drug-target pair based on ChEMBL (1 for drugs, 2 target and 0 for neither); (2) sequential label marking the target and all of the drug names appearing; (3) single label indicating whether the sentence contains a drug-target pair (1 for "Good" and 0 for "Other"). See Figure 1 as a example of labeling.

Sentence	In	contrast,	triclosan	does	not	increase	the	binding	of	NAD+	to	Fabl	(G93V).
type (1) label	0	0	1	0	0	0	0	0	0	0	0	2	0
type (2) label	0	0	1	0	0	0	0	0	0	1	0	2	0
type (3) label							1						

Figure 1: Three types of label of a "Good sentence" sample

Note that the final ensemble model takes type (1) label as its targeted label, while the NER component optimizes its prediction towards type (2) label, and the sentence classifier is trained on type (3) label.

One important observation is that ChEMBL only collects a subset of all eligible target-drug pairs from their list of reference documents. So therefore, precision is not a very meaningful metric to evaluate the final ensemble model.

3 Methodology

Our final model combines an attention Recurrent Neural Network (RNN) for sentence classification and an improved NER model. This two parts are trained with the same training set but independently. For each input sentence on the testing set, if the classifier determines that it contains a drug-target pair, i.e. it is classified as a "Good sentence", then the NER model will be applied to extract drug names. Otherwise, we will consider there exists no eligible entities.

3.1 Sentence classifier

The attention RNN sentence classifier consists of the following major components: word embedding, sequence encoder, composition layer and the toplayer classifier. Figure 2 shows a view of the architecture of our neural network.

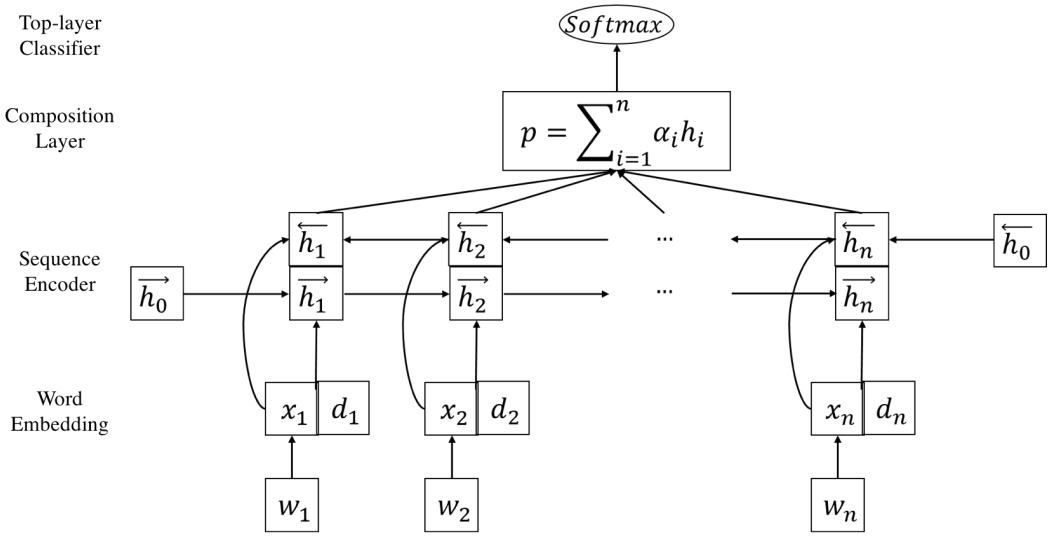

Figure 2: Attention Neural Network classifier

Given a sentence with words $(w_1, w_2, ..., w_n)$, where n is the length of the sentence, we first embed the words to vectors, either through an learned embedding matrix W_e or by the pre-trained word2vec. We concatenate the word embedding with additional information d_i of its Part-of-Speech(POS) tagging and dependency relations, which are obtained by the pre-trained Parsey McParseface parser (*SyntaxNet*[2]). Also, d_i contains a target indicator, i.e. 1 if w_i is the target and 0 if not. In this way, the neural network is informed where the target is in each input sentence. We then use a bidirectional Long Short-Term Memory(LSTM) or Gated Recurrent Unit(GRU) to get annotations of words by summarizing information from both direction of words, and therefore incorporate the contextual information in the annotation.

The attentive composition layer (Lin et al., 2017)[1] uses an attention mechanism over the hidden states of a bidirectional Recurrent Neural Network (bi-RNN) to generate a representation p of an input sentence. The attention mechanism is defined as

$$u_i = tanh(W_w h_i + b_w) \quad (1)$$

$$\alpha_i = \frac{exp(u_i^T u_w)}{\sum_i u_i^T u_w} \quad (2)$$

$$p = \sum_i \alpha_i h_i \quad (3)$$

where $h_1, h_2, ..., h_n$ are the output hidden vectors of a bi-RNN. These are fed to an affine transformation (W_w, b_w) which outputs a set of keys $u_1, u_2, ..., u_n$. The α_i represent the score of similarity between the keys and a learned context query vector

[2]https://github.com/tensorflow/models/tree/master/research/syntaxnet

u_w. These weights are used to produce the final representation p, which is a weighted linear combination of the hidden vectors.

The purpose of word attention layer is to identify which of the words it needs to attend over to determine whether the sentence contains any drugs related to the specified target. For example, in a "Good sentence" like *"We suggest that felbamate interacts with a unique site on NR2B..."*, where "felbamate" is a drug and "NR2B" is a target, we would expect the attention RNN to focus on words "interacts with (a unique) site on", as they form a predicate linking the drug-target pair. Therefore, the classifier could help the final model reduce false positive errors.

3.2 NER

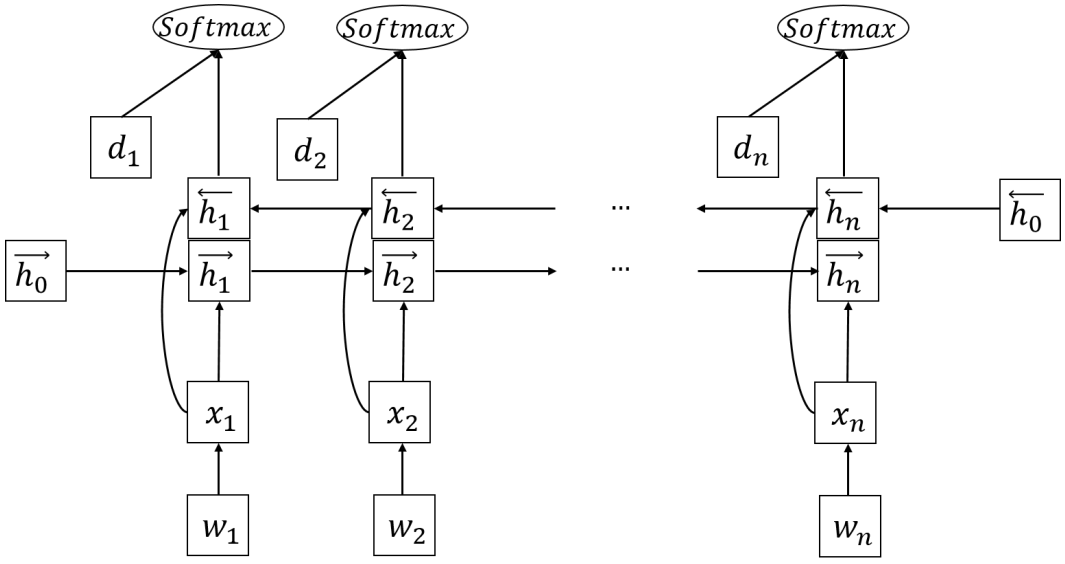

Figure 3: NER model with the 2nd type of configuration

The NER model is a many-to-many bi-RNN model: it takes a sentence as input, and for each word in the sentence outputs its probabilities of being a drug name(1), a target name(2) or neither(0). As $\{d_i\}$ contains target indicators, predictions of the target class is trivial.

The 1st model configuration (refer to Figure 2) feeds the concatenation of word embeddings and additional information (POS tagging, dependency relations and target indicators) to bi-RNN. The 2nd configuration (Figure 3) incorporates these additional information into the top-layer softmax functions.

4 Results

4.1 Sentence classifier

We compare the attention RNN sentence classifier with several baselines, including fastText model (Joulin et al., 2016)[2], unidirectional RNN model, and bidirectional RNN model without attention layer.

FastText is a simple but efficient linear method. It represents each sentence by averaging the word-embedding vectors.

Unidirectional RNN model summarizes information of predicates from only one direction. We feed the last hidden vector to a softmax function. However, this model takes quite along time to train, as its loss or accuracy might hardly converge, which results from the fact that the last hidden unit of a uni-RNN could not well capture the semantic of the whole sentence, especially when the sequence is rather long.

Bidirectional RNN without attention reads each sentence both forwards and backwards. The hidden vectors are then averaged into a sentence representation without weights. This model could be viewed as a degenerate case of the attention-based model, when the attention mechanism returns a constant regardless of its input.

#	Model	Dev	Test
1	FastText(word2vec)	72.3	70.4
2	AttnBiGRU	76.3	74.6
3	BiLSTM(word2vec)	78.7	78.0
4	AttnBiLSTM(word2vec, dropout)	79.9	79.2
5	BiGRU(word2vec)	81.0	79.3
6	AttnBiGRU(word2vec)	81.5	79.8
7	AttnBiGRU(word2vec, dropout)	80.6	**81.0**

Figure 4: Accuracy[%] on development and testing set with same parameters

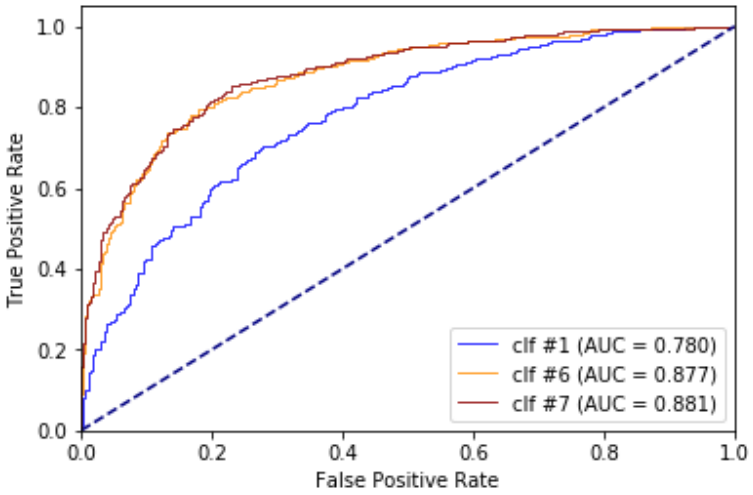

Figure 5: Receiver Operating Characteristic (ROC) curves of selected classifiers

Note that all models in Figure 4 are trained with early stop mechanism, i.e. to stop training if the fluctuation of accuaracy on development set is smaller than a certain threshold. It should also be noted that the base rate (the proportion of positives) is 53.1%.

As shown in Figure 4 and Figure 5, clf `#7`, i.e. AttnBiGRU using the pre-trained word2vec and recurrent dropout (Semeniuta et al., 2016)[3], gives the best performance. Generally speaking, models using word2vec perform better than those without a pre-trained word embedding. Also, GRU models achieve better accuracies than LSTM overall.

Furthermore, as expected, to determine the label as a "Good sentence", the attention RNN classifier assigns much more attentive weights to the those words indicating the relationship between the drug the target, for instance, "interacts with", "inhibit". Instead of capturing the semantic of the whole sentence, it is sufficient for the classifier to pay more attention to the predicates which link the drug-target pairs. See Figure 6 for attention distributions of "Good sentence" samples.

By contrast, "Other sentence" usually does not contain any predicate which explicitly indicates the existence of a related drug in the sentence. So therefore, as shown Figure 7, the attentive weights are rather averaged. Note that in order to have a fixed length input for each batch, the sequences are padded with zero vectors to fill up the remaining time steps in the batch. The paddings would also be assigned attentive weights, which do not show in Figure 6 and Figure 7.

We	suggest	that	felbamate	**interacts**	**with**	a	unique	site	on	NR2B	...	
0.04	0.79	0.16	0.06	**81.89**	**13.91**	0.02	2.21	0.27	0.43	0.00	...	
These	results	characterize	SL65.0155	as	a	novel	promnesic	**agent**	**acting**	**via**	5-HT(4)	...
0.01	0.06	0.02	0.01	6.51	0.11	9.35	0.10	**48.40**	**21.27**	**12.56**	0.01	...
PI-88	**inhibited**	the	decrease	**in**	**levels**	of	FGF-2	protein	...			
0.05	**50.92**	1.69	1.28	**35.71**	**3.83**	0.02	0.03	0.05	...			
CG100649	**significantly**	**inhibits**	CA-II	in	blood	and	COX-2	in	inflammatory	tissue.		
0.08	**69.95**	**23.44**	0.01	3.68	0.04	0.00	0.00	2.46	0.02	0.01		

Figure 6: Normalized attention distribution[%] on "Good sentence" samples, where drug names are highlighted in yellow and targets are highlighted in pink

COX-2	protein	levels	correlates	with	RNA	expression.		
0.12	0.18	0.64	0.29	0.22	0.11	0.23		
However,	functional	data	regarding	a	renal	SUR	are	lacking.
0.29	0.39	0.73	0.45	0.10	0.45	0.91	1.42	2.50

Figure 7: Normalized attention distribution[%] on "Other sentence" samples, where targets are highlighted in pink

		Actual	
		p	n
Predicted	Y	336	88
	N	84	396

Figure 8:
Confusion matrix of clf #7

From the confusion matrix of clf `#7` when threshold = 0.5 (Figure 8), its sensitivity and specificity are rather balanced. We also analyze the false positive and false negative errors. One of the most underlying reason accounting for making false negative errors is that, there is no predicate or phrase which directly connects the drug-target pair appearing explicitly in the sentence. For example, *"We also explored the interactions of NGF with its natural receptors, TrkA and P75, and how tanezumab blocks them."*, where "NGF" is a target and "tanezumab" is a drug. In this case, the relation between "tanezumab" and "TrkA and P75" can easily be captured, while there is not any obvious word indicating the relation between "tanezumab" and "NGF".

As for false positive errors, some of them occur due to the limitation of ChEMBL dataset. Take a false positive error as example: *"The mAb 5F11 showed specific binding to CD30"*, where the target is "CD30". Even though "mAb 5F11" is actually a drug name, we fail to label it as a "Good sentence" because "mAb 5F11" is not in ChEMBL's drug name list.

4.2 NER

Recall, precision and their harmonic mean, i.e. the F-measure, are three commonly-used evaluation metrics for NER problems. They are defined as:

$$\text{Recall} = \frac{\#\text{drugs an NER correctly detected}}{\#\text{drugs contained in the input text}} = \frac{\text{TP}}{\text{TP+FN}}$$

$$\text{Precision} = \frac{\#\text{drugs an NER correctly detected}}{\#\text{drugs identified by the NER}} = \frac{\text{TP}}{\text{TP+FP}}$$

$$\text{F-measure} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision+Recall}}$$

When the number of false negative errors (i.e. the drugs which are actually in the document but not detected) decreases, the recall will go up. When the number of false positive errors (i.e. the drugs which are wrongly detected) decrease, the precision will go up.

We experiment with the pre-trained word2vec embedding, different model configurations and recurrent dropout.

#	RNN type	word2vec	config.	dropout	Test Recall	Test Precision	Test F-measure
1	LSTM	1	2	0.5	68.2	72.9	70.5
2	GRU	1	1	0.5	66.8	74.9	70.6
3	GRU	0	1	0	72.5	69.6	71.0
4	LSTM	0	1	0	**74.0**	72.2	73.1
5	LSTM	0	1	0.5	67.3	80.0	73.1
6	GRU	0	2	0.5	65.9	83.6	73.7
7	LSTM	0	2	0	66.2	85.5	74.6
8	GRU	0	1	0.5	68.3	82.5	74.8
9	LSTM	0	2	0.5	68.3	84.8	**75.7**

Figure 9: Recall[%] and precision[%] on testing set with same parameters ("word2vec" column shows 0 means using a learned word embedding, 1 means using the pre-trained word2vec)

Note that all models in Figure 9 are trained with early stop mechanism, i.e. to stop training if the fluctuation of recall on development set is smaller than a certain threshold. Figure 9 shows that NER models using the 1st configuration without recurrent dropout gives relatively higher recall, while ner `#6` gives the highest F-measure. In general, models using learned word embedding and regulizations perform better.

4.3 Ensemble model

Since the classifier excludes part of the detected drugs, the model ensembling will increase the overall precision (i.e. we identify more drugs that are correctly paired with a corresponding target) but decrease the recall (i.e. we may be missing drug-target pairs), we choose the NER models with the recall larger than 70%: ner `#3` and ner `#4`.

Our baseline models are traditional NER models (using BiRNNs) trained with type (1) label directly, without any POS taggings or dependency relations as input.

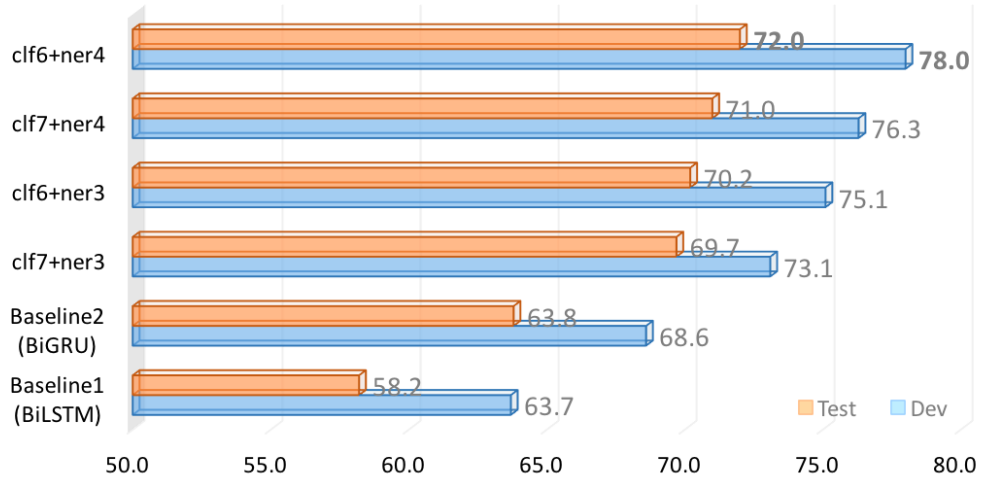

Figure 10: Recalls[%] of the ensemble models on development and testing set

Figure 10 shows the recalls of selected ensemble models compared with baseline models, showing the superiority ($\approx +8\%$ in recall) of our novel ensemble approach. The combination of clf `#6` and ner `#4` gives the highest recall. And we find that there is a drop in recall for each ensemble model, compared with their pure NER components.

5 Conclusion and Forthcoming Research

We developed an ensemble neural networks approach to identify drug names acting on the targets of interest from text data. The attention-based sentence classifier helps filter the detected drug-target pair whose relationship cannot be justified. We

also incorporated the POS tagging and dependency relations of each input sentence into RNN models, so that the models could better understand the sentence structure.

Moreover, we tested the model in the wild. See appendix for the deployment results of our NER on the target named "JAK". It shows that there exists quite a few noises in the predictions, as the model mistakes a target name as a drug name from time to time.

There is always a fundamental trade-off between recall and precision when it comes to NER problems. For the purpose of maximizing our NER capability of capturing drug entities, the model is trained and selected with more weight on recall, but at the cost of a drop in precision. One possible solution to this issue is to ensemble another drug name classifier on the top layer, based on Natural Language Inference (NLI) methodologies. NLI could help further justify the relations of drug-target pairs in a given context.

Furthermore, to better capture the eligible drug-target pairs with strong relations, another method is to identify drugs based on the dependency parser output. With extracting the predicates or phrases linking two entities in the sentence, a predicate classifier could be built to determine whether it indicates a strong relation between the subject and object in the sentence.

6 Acknowledgements

I would first like to thank my instructors Yoann Mamy Randriamihaja and Bill McMahon at Roivant. They were always open whenever I ran into a trouble spot or had a question about my research. They consistently allowed this project to be my own work, but steered me in the right direction whenever they thought I needed it. Yoann was also the second reader of this paper and the capstone poster, and I am gratefully indebted to him for his very valuable comments on this writing.

I would also like to thank my advisor Michael Gill at Center for Data Science, NYU. I decided to switch to this topic at a rather late time. He supported me to work on what I am truly interested in, and provided quite a few valuable suggestions for my project proposal.

Finally, I must express my profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study.

References

[1] Z Lin, M Feng, CN Santos, M Yu, B Xiang,B Zhou and Y Bengio. A structured self-attentive sentence embedding. *International Conference on Learning Representations*. 2017. 3.1

[2] A Joulin, E Grave, P Bojanowski and T Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*. 2016. 4.1

[3] S Semeniuta, A Severyn and E Barth. Recurrent dropout without memory loss. *arXiv:1603.05118 [cs.CL]*. 2016. 4.1

Appendix: Deployment Samples

Explanation: We randomly select 10 samples of drug name prediction for the target "JAK". Samples itemized with tick symbols correctly detect the drug names, while samples itemized with cross symbols are errors. Within each sentence, targets are highlighted in pink, and the detected drug names are highlighted in yellow.

× However, Ob-Ra, which contains a truncated intracellular domain, and unable to activate the STAT pathway [41] may still transduce signals by way of activation of JAK2, IRS-1 or extracellular factor-regulated kinases (ERKS) including MAPKs (mitogen-activated protein kinases) [42].
- Note: "IRS-1" is a target name.

× OPB-31121 is a novel STAT3 inhibitor that selectively inhibits IL-6-dependent phosphorylation of STAT3 without inhibiting JAK2 phosphorylation [45].
- Note: "OPB-31121" is a drug inhibiting "STAT3", but not "JAK2"

✓ JSI-124 was identified as a JAK2 inhibitor that suppresses STAT3 phosphorylation but displays much lower activities toward JAK1 and src (25).

× Compared with LG group, the expressions of JAK2, p-STAT1, p-STAT3,and TGF-beta1, mRNA were significantly increased in HG group from 6 to 72hours.
- Note: both "p-STAT3" and "TGF-beta1" are target names.

✓ By contrast, we observe that AG490, a JAK family-selective inhibitor, anddominant negative Jak1 protein can significantly inhibit Stat3-induced DNAbind-

ing activity as well as Stat3-mediated gene activation in NIH3T3 cells.

× To functionally test the role of STAT3 signaling in these mice, we used cucurbitacin I, a selective JAK2 /STAT3 pharmacological inhibitor (Blaskovich et al., 2003).

- Note: Blaskovich is the author of a cited paper.

✓ As a pan-JAK inhibitor, 6BIO also inhibits phosphorylation of SRC in human melanoma cells.

✓ Ruxolitinib , a JAK-1/2 inhibitor, is effectiveat controlling splenomegaly and constitutional symptoms, but has limitedbenefit in reversing bone marrow fibrosis or inducing complete or partialremissions.

✓ Efficacy and Safety of ABT-494 , a Selective JAK-1 Inhibitor, in a Phase IIb Study in Patients With Rheumatoid Arthritis and an Inadequate Response to Methotrexate.

✓ To test whether G5-7 also inhibits the phosphorylation of STAT3 by JAK2 , we performed in vitro kinase assay with purified JAK2 and STAT3 proteins.