Multi-input Recurrent Independent Mechanisms for leveraging knowledge sources: Case studies on sentiment analysis and health text mining

Parsa Bagherzadeh and Sabine Bergler
CLaC Labs, Concordia University
Montréal, Canada
{p_bagher, bergler}@cse.concordia.ca

Abstract

This paper presents a way to inject and leverage existing knowledge from external sources in a Deep Learning environment, extending the recently proposed Recurrent Independent Mechnisms (RIMs) architecture, which comprises a set of interacting yet independent modules. We show that this extension of the RIMs architecture is an effective framework with lower parameter implications compared to purely fine-tuned systems.

1 Introduction

Deep neural networks have been successfully applied to a variety of natural language processing tasks such as text classification, sequence labeling, sequence generation, etc. Deep architectures are often non-modular, homogeneous systems and trained end-to-end. End-to-end training is performed with the hope that the structure of a networks is sufficient to direct gradient descent from a random initial state to a highly non-trivial solution (Glasmachers, 2017).

An important issue with the end-to-end training is that throughout the training of a system composed of several layers, valuable information contained in a problem decomposition that resulted in a specific network design is ignored (Glasmachers, 2017). In non-modular systems, explicit decomposition of high level tasks into distinct subprocesses is not possible and necessary complexity has to be induced through the complexity of the input stimulus. This results in large systems whith the required number of training samples becoming intractable. Interpretation of these black box systems is difficult (Miikkulainen and Dyer, 1991).

In compositional systems, in contrast, smaller modules encode specialized expertise which is known to impact one aspect of the task at hand. The aggregation of the modules acts synergistically to address the overall task. In a modular system, the components act largely independently but communicate occasionally. Module autonomy is crucial because in the case of distributional shifts (significant changes in some modules), other modules should remain robust (Schölkopf et al., 2012), (Goyal et al., 2019). Modules also need to interact occasionally to achieve compositional behavior (Bengio, 2017).

Many current neural modular systems, such as EntNet (Henaff et al., 2017) and IndRNN (Li et al., 2018), offer only module independence, but no module communication. The recently proposed Recurrent Independent Mechanisms (RIMs) (Goyal et al., 2019), however, suggest to model a complex system by dividing the overall model into M communicative recurrent modules.

Deep architectures often rely solely on raw data in large quantities with a requirement of representativeness regarding task requirements. This becomes problematic for tasks with a specialized, low-frequency terminology, where high quality knowledge sources for NLP and AI are often available and have proven their effectiveness. Embedding expert knowledge in extended pre-trained word embeddings is costly. We present untied inedpendent modules to embed knowledge from different sources onto systems input. Knowledge sources, as independent experts, provide different annotations (abstractions) for the input, combining various classifications for solving the task.

For instance, providing sentiment lexica for sentiment analysis reduces the demand for training data by expanding the limited training vocabulary with an extended set of annotated terms. Precompiled word embeddings are to be considered knowledge sources in the same spirit and we demonstrate that they inter-operate with a variety of other knowledge sources such as gazetteers and POS encoding.

Consider Example 1 from the Stanford Sentiment Treebank (SST-2) (Socher et al., 2013).

(1) *This is an absurd comedy about alienation, separation and loss.*

Proceedings of Deep Learning Inside Out (DeeLIO):
The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures, pages 108–118

Figure 1 shows annotations from different knowledge sources for Example 1, such as tokenization (from the ANNIE tokenizer), POS tags (from the Stanford POS tagger), and sentiment annotations from three sentiment lexica (AFINN (Nielsen, 2011), MPQA (Wilson et al., 2005), and NRC (Mohammad et al., 2013)).

	x_t^1	x_t^2	x_t^3	x_t^4	x_t^5
t	Token	POS	AFINN	MPQA	NRC
1	This	DT	0	Neutral	-0.19
2	is	VBZ	0	Neutral	0.00
3	an	DT	0	Neutral	0.08
4	absurd	JJ	0	Neg.	-1.56
5	comedy	NN	+1	Neg.	0.27
6	about	IN	0	Neutral	-0.34
7	alienation	NN	-2	Neg.	0.00
8	,	,	0	Neutral	0.27
9	separation	NN	0	Neutral	-0.29
10	and	CC	0	Neutral	0.41
11	loss	NN	-3	Neg.	-0.51
12	.	.	0	Neutral	-0.06

Figure 1: Various annotations for Example 1

The annotations of the different sentiment lexica in Figure 1 vary substantially: *comedy* is classified as positive (+1) in AFINN, as negative in MPQA, and almost neutral in NRC. (Özdemir and Bergler, 2015a) showed that this variance in judgements is not prohibitive, in fact (Özdemir and Bergler, 2015b) showed that combining 5 sentiment lexica outperformed all other combinations. These differences are in fact advantageous in an ensemble setting and reflect diversity among experts. The differences cannot be exploited, when a single embedding is used for tokens, but may be retained, when different lexica are embedded independently in different modules.

We add input independence to the RIMs architecture, providing different language annotations as inputs to a set of independent, but interacting modules. The resulting system is a flexible modular architecture for leveraging token-level knowledge in form of different annotation embeddings, which will be given different weights for the task at hand dependeing on their usefulness during training (see Figure 11). The system is evaluated on tasks such as sentiment analysis and analysis of health-related tweets for different health concerns.

Our experiments demonstrate that leveraging knowledge sources under a modular framework consistently improves performance with little increase in parameter space. Additionally, when frozen language models are supplemented with knowledge sources, the drop in performance is minimal, making this technique particularly beneficial for users that do not have access to powerful computational resources. Lastly, the modular nature of the system allows to visualize the models functionality.

2 Methods

2.1 RIMs

Recurrent independent mechanisms (RIMs) is a modular architecture that models a dynamic system by dividing it into M recurrent modules (Goyal et al., 2019). At time-step t, each module R_m ($m = 1, \ldots, M$) has a hidden state $h_t^m \in \mathbb{R}^{d_h}$.

Input selection Each module R_m gets the augmented input $X_t = x_t \oplus \mathbf{0}$, where $\mathbf{0}$ is an all-zero vector and $\oplus$ is the row-level concatenation. Then, using an attention mechanism, module R_m selects input:

$$A_t^m = softmax(\frac{h_{t-1}^m W_m^{query} (X_t W^{key})^T}{\sqrt{d}}) X_t W^{val} \quad (1)$$

where $h_{t-1}^m W_m^{query}$ is the *query*, $X_t W^{key}$ is the *key*, and $X_t W^{val}$ is the *value* in the attention mechanism (Vaswani et al., 2017). The matrices $W_m^{query} \in \mathbb{R}^{d_h \times d_{in}^{query}}$, $W^{key} \in \mathbb{R}^{d_{in} \times d_{in}^{key}}$, and $W^{val} \in \mathbb{R}^{d_{in} \times d_{in}^{val}}$ are linear transformations for constructing query, key, and value for the input selection attention.[1]

If the input x_t is considered relevant to module R_m, the attention mechanism in Equation 1 assigns more weight to it (selects it), otherwise more weight will be assigned to the null input (Goyal et al., 2019).

The $softmax$ values of Equation 1 determine a set S_t of top m_{Active} modules.[2] Among M modules, those with the least attention on the null input are the active modules. The selected input A_t^m determines a temporary hidden state $\tilde{h}_t^m$ for the active modules:

$$\tilde{h}_t^m = R_m(h_{t-1}^m, A_t^m) \quad m \in S_t \qquad (2)$$

where $R_m(h_{t-1}^m, A_t^m)$ denotes one iteration of updating the recurrent module R_m based on previous state h_{t-1}^m and current input A_t^m. The hidden states

[1] d_{in}^{query}, d_{in}^{key}, and d_{in}^{val} are dimensionalities of query, key, and value respectively (for the input selection attention)

[2] The cardinality $|S_t| = m_{Active}$ is currently a fixed hyperparameter, that can ultimately be determined based on the target task.

of the inactive modules R_m ($m \notin S_t$) remain unchanged:

$$h_t^m = h_{t-1}^m \quad m \notin S_t \tag{3}$$

Module communication To obtain the actual hidden states h_t^m, the active modules communicate using an attention mechanism:

$$h_t^m = softmax(\frac{Q_{t,m}(K_{t,:})^T}{\sqrt{d_h}})V_{t,:} + \tilde{h}_t^m \quad m \in S_t \tag{4}$$

where

$$Q_{t,m} = \tilde{h}_t^m \tilde{W}_m^{query}$$

$K_{t,:}$ is the row-level concatenation of all $K_{t,m}$ ($m = 1, \ldots, M$) defined as:

$$K_{t,m} = \tilde{h}_t^m \tilde{W}_m^{key}$$

and $V_{t,:}$ is the row-level concatenation of all $V_{t,m}$ ($m = 1, \ldots, M$) defined as:

$$V_{t,m} = \tilde{h}_t^m \tilde{W}_m^{val}$$

The matrices $\tilde{W}_m^{query} \in \mathbb{R}^{d_h \times d_{com}^{query}}$, $\tilde{W}_m^{key} \in \mathbb{R}^{d_h \times d_{com}^{key}}$ and $\tilde{W}_m^{val} \in \mathbb{R}^{d_h \times d_{com}^{val}}$ are used for constructing query, key, and value for the *communication* attention.[3]

Note that both the key $K_{t,:}$ and the value $V_{t,:}$ depend on the temporary hidden states of all modules, therefore h_t^m in Equation 4 is determined by attending to all modules. The overall hidden state of the RIMs model at time-step t can be defined as $h_t = [h_t^1, \ldots, h_t^M]$ which is the concatenation of the hidden states of all modules.

Classification We choose a simple attention layer together with a classifier to obtain the appropriate vector representation of a given sample. Attention (Bahdanau et al., 2015) determines importance scores $e_t = w_{att}^T h_t$ using a latent context vector w_{att}. The score is then normalized using $\alpha_t = \frac{exp(e_t)}{\sum_j e_j}$ for a weighted sum $H = \sum_t \alpha_t * h_t$, which is the input for a classifier.

2.2 Multi-input RIMs

We extend this architecture to so-called *multi-input RIMs*, which consist of a set of M modules, similar to the standard RIMs. The standard RIMs model assumes the same input sequence for all modules (X_t in Equation 1), which share the same linear transformation matrices W^{key} and W^{val} for constructing the keys and values for the attention mechanism.

[3] d_{com}^{query}, d_{com}^{key}, and d_{com}^{val} are dimensions of query, key, and value respectively (for the *communication* attention)

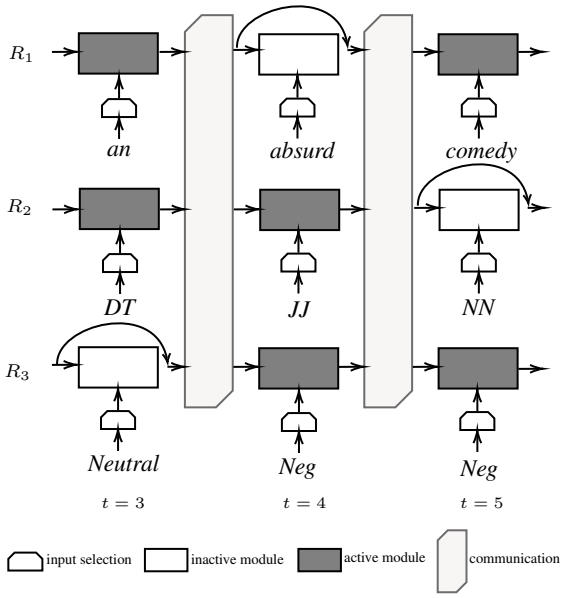


Figure 2: A 3 module multi-input RIMs for Example 1 at $t = 3, ..., 5$. The dynamics of each module is independent of the others and active modules communicate at each time-step

In contrast, we untie the input attention mechanism and consider dedicated linear transformations W_m^{key} and W_m^{val} for module R_m. Untying the attention mechanism allows modules to have different inputs X_t^m ($m = 1, \ldots, M$) each potentially with a different dimensionality. This supports our use of each module to encode a different knowledge source, one being word embeddings, one being a gazetteer list, etc. The input selection mechanism of Equation 1 then expands to Equation 5:

$$A_t^m = softmax(\frac{h_{t-1}^m W_m^{query} (X_t^m W_m^{key})^T}{\sqrt{d_h}}) X_t^m W_m^{val} \tag{5}$$

where $X_t^m = x_t^m \oplus \mathbf{0}$.

In Equation 5, the $softmax$ produces two attention scores, i.e. how much the module R_m attends to the input x_t^m and the null input $\mathbf{0}$. The top m_{active} modules with least attention scores to the null input form a set S_t. The temporary hidden state for active modules is determined by Equation 2 and modules communicate according to Equation 4, identical to standard RIMs. An illustration of the multi-input RIMs model is provided in Figure 2.

3 Tasks

We explore the potential of multi-input RIMs by ablation on different tasks that are each very specific in their description and do not have large training datasets, namely three sentiment analysis tasks and

two health-related tweet classification tasks.

3.1 Sentiment analysis

Here we consider three sentiment benchmark datasets with their respective tasks:

SST-2 Stanford sentiment tree-bank for the task of binary sentiment classification of movie reviews (Socher et al., 2013). The models are trained on the data provided by the GLUE benchmark[4] (Wang et al., 2018).

SE17-4A SemEval 2017 task 4 subtask A is a 3-class problem for sentiment classification of tweets (Rosenthal et al., 2017). The tweets are classified as *Negative*, *Neutral*, and *Positive*. The performance for this task is measured by the macro-average of recall scores for positive, negative, and neutral classes and evaluated by the TweetEval benchmark (Barbieri et al., 2020)[5].

SE15-11 SemEval 2015 task 11 is a pilot task of sentiment analysis for figurative language tweets. The training set comprises a collection of sarcastic, ironic, and metaphoric tweets (4490 tweets) annotated on an 11 point scale $(-5, \ldots, +5)$ (Ghosh et al., 2015). The performance is measured by *Cosine* similarity between the gold standard labels and predictions.

We use the following sentiment lexica as knowledge sources:

1. *AFINN*: A manually compiled lexicon of 2500 words, rated for valence scores with an integer between -5 and 5 together with their prior polarities (Nielsen, 2011).

2. *MPQA*: A manually compiled lexicon of 8000 words, distinguishing positive, negative, and neutral sentiment scores (Wilson et al., 2005).

3. *NRC HashTag sentiment*: An automatically compiled resource, that uses seed hashtags (Mohammad et al., 2013). The polarity of the seed hashtag is used to calculate PMI-based[6] scores (Church and Hanks, 1990).

The training set SE15-11 has been released as tweet IDs and part of the training set is not available anymore[7], therefore we randomly select 20% of the available tweets as test set and use the remaining for training.

3.2 Health experience classification of tweets

Personal experiences gleaned from social media can enhance awareness of the state of public health. Here we focus on two tasks:

SM18-2 The task of medication intake report detection was introduced as SMM4H 2018 Task 2 (Weissenbacher et al., 2018) as a 3-way classification task. Tweets in which the user clearly expresses a personal medication intake/consumption are considered Class 1. Tweets where the user may have taken some medication are labeled as Class 2. Class 3 tweets mention medication names but do not indicate personal intake. The total number of samples in the training set is 17700.

SM20-5 Birth defect mention detection concerning a child is a 3-class problem, where Class 1 tweets indicate that the user's child has a birth defect. Class 2 tweets are unclear as to whether the poster speaks of birth defects of their child. Class 3 tweets merely mention birth defects but not with respect to the poster's child (Klein et al., 2020). The training set includes 18382 samples.

Both, SM18-2 and SM20-5 benefit from specialized gazetters of relevant medical terms, in particular:

1. *Drugs*: A gazetteer list of drug names compiled from Drug Bank (Wishart et al., 2018).

2. *Diseases*: A list of terms for *infections, wounds, injuries, pain, etc.*, compiled from subtree C in MeSH[8] (Lipscomb, 2000). Disease mentions are important evidence for medication intake classification, since drugs are usually consumed to treat a disease.

3. *Birth Defect*: Congenital, hereditary, and neonatal diseases and abnormalities (from MeSH C16).

[4] http://gluebenchmark.com

[5] https://github.com/cardiffnlp/tweeteval

[6] point-wise mutual information

[7] about 33% of the tweets are not available

[8] https://meshb.nlm.nih.gov/treeView

4. *Pregnancy*: Pregnancy complication terms (from MeSH C13.703)

For SM18-2, the gold labels of the competition set have not been disclosed, therefore we randomly hold out a test set (20% of the original training data). For SM18-2 and SM20-5 the performances are measured in terms of micro-F1 scores for 0 and 1 class.

4 Implementation

Preprocessing We preprocess the data using a GATE pipeline (Cunningham et al., 2002) with the ANNIE English Tokenizer (for SST-2 task) and ANNIE tweet tokenizer as well as the hashtag tokenizer (for the tweet tasks).

Embeddings Each annotation type provides a sequence (see Figure 1) which is used as input for a dedicated module in multi-input RIMs. Therefore, each sequence has to be properly embedded. The annotation types can be embedded either using pretrained embeddings or using randomly initialized embeddings that are learned during the training.

Tokens are embedded using ELMo (Peters et al., 2018) or RoBERTa (Liu et al., 2019) pretrained models. For ELMo, we use the pretrained model provided by AllenNLP[9] and for RoBERTa, the model provided by Hugging Face[10].

POS tags following (Bagherzadeh and Bergler, 2021), we apply Word2Vec on POS tag sequences instead of token sequences. The POS embeddings are trained using the Gensim package (Rehurek and Sojka, 2010) with a window size of 5 and dimensionality 20. The pretraining is performed on combined training data of all tasks introduced in Section 3.

AFINN and NRC matches do not require an embedding, since the lexica quantify the sentiment scores numerically.

MPQA matches for *Negative*, *Neutral*, and *Positive* polarities are encoded numerically by -1, 0, and 1 respectively.

Medical Gazetteer matches are embedded using a learnable embedding matrix $B \in \mathbb{R}^{5\times 20}$. The 5 rows in B correspond to 4 medical resources[11] plus one row to indicate no annotation.

The multi-input RIMs model is a flexible architecture and the modules can be of any recurrent type. Here, we use LSTMs for complex inputs, such as *Token* or *POS*, and RNNs for annotations with simpler encodings, such as gazetteers.

Module	d_{in}	d_h	d_{in}^{quer}	d_{in}^{key}	d_{in}^{val}	d_{com}^{quer}	d_{com}^{key}	d_{com}^{val}
Token	1024	256	512	512	1024	64	64	256
POS	50	256	100	100	50	64	64	256
Senti[1]	1	256	16	16	1	64	64	256
Medic[2]	20	256	100	100	20	64	64	256

1: AFIIN, MPQA, NRC
2: Drug, Preg, BirthDef, Disease

Figure 3: Hyper-parameters used in the experiments.

Figure 3 summarizes the hyper-parameters used for multi-input RIMs. We use the learning rates of $lr = 0.5e-2$ and $lr = 0.5e-4$ for ELMo- and RoBERTa-based models respectively. The hyper-parameters are tuned based on a grid-search approach. The multi-input RIMs model itself (excluding the language models) has 4M learnable parameters.

To calculate classification loss we use cross-entropy loss and we optimize the models using the Adam optimizer (Kingma and Ba, 2015). The models are implemented using PyTorch (Paszke et al., 2017).

5 Numerical results

We present a set of ablation studies to evaluate the effectiveness and contribution of different knowledge sources.

All modules active Figures 4–6 report results for the multi-input RIMs model when the modules are provided with different annotation types and all modules are kept active ($M = m_{Active}$). For the runs where the *Token* annotation is the only input ($M = 1$), the model is reduced to a simple LSTM with ELMo or RoBERTa embeddings, which we consider to form baselines.

Figure 4 shows that all sentiment tasks benefit from the sentiment lexica. For SST-2, *AFINN* and *MPQA* add more to the task than *NRC*. On the other hand, *NRC* yields considerable performance improvements for the tweet sentiment data sets of

[9] https://allennlp.org/
[10] https://huggingface.co/
[11] Drug, Disease, Birth defect, and Pregnancy

		SST-2 (Acc %)		SE17-4A (mac-Rec %)		SE15-11 (Cosine)	
M	Annotations	ELMo	RoBERTa	ELMo	RoBERTa	ELMo	RoBERTa
1	Token	88.5	96.4	64.1	70.2	78.1	82.2
2	Token + AFINN	91.2	96.7	66.8	71.6	80.1	83.0
2	Token + MPQA	90.3	96.5	65.9	71.2	80.0	83.2
2	Token + NRC	89.7	96.4	67.1	71.5	82.1	83.9
2	Token + POS	89.2	96.4	65.2	70.8	78.9	82.2
3	Token + POS + AFINN	91.8	97.1	68.3	72.0	81.4	83.3
3	Token + POS + MPQA	90.7	96.9	67.2	71.8	81.1	83.3
3	Token + POS + NRC	90.5	96.5	68.9	72.4	82.6	84.4
5	Token + POS + AFINN + MPQA + NRC	**92.3**	**97.3**	**70.4**	**73.3**	**83.2**	**85.0**
1	Token$^{\mathcal{F}}$	83.2	94.1	61.1	68.2	75.3	80.3
5	Token$^{\mathcal{F}}$ + POS + AFINN + MPQA + NRC	89.1	95.4	68.2	71.6	81.4	84.1

$\mathcal{F}$: Frozen language model

Figure 4: Multi-input RIMs on sentiment tasks with knowledge sources. Each annotation is the input of a dedicated module. In each run, all modules are kept active ($m_{Active} = M$)

SE17-4a and SE15-11. We surmise the greater effectiveness of the *NRC* lexicon for the tweet sentiment tasks is due to the fact that it is constructed from tweet corpora.

POS constitutes general linguistic knowledge and demonstrates consistent yet small improvements for the sentiment tasks. However, *POS* improves performance for the health concerns data of SM18-2 (Figure 5) and SM20-5 (Figure 6). Note that both tasks concern detection of personal experience mentions, for which categories such as pronouns (both personal and possessive) and verbs in past tense are important, which carry distinctive POS tags.

POS constitutes general linguistic knowledge and demonstrates consistent yet small improvements for the sentiment tasks. However, *POS* improves performance for the health concerns data of SM18-2 (Figure 5) and SM20-5 (Figure 6). Note that both tasks concern detection of personal experience mentions, for which categories such as pronouns (both personal and possessive) and verbs in past tense are important, which carry distinctive POS tags.

Improvements from medical knowledge gazetteers are also compelling. Figure 5 shows that the *Disease* gazetteer enhances the performance for the medication intake task, corroborating the hypothesis that disease mentions are strong evi-

M	Annotations	ELMo	RoBERTa
1	Token	68.2	72.0
2	Token + Drug	71.3	73.9
2	Token + Disease	70.5	73.0
2	Token + POS	71.5	74.1
3	Token + POS + Drug	73.6	74.8
3	Token + POS + Disease	72.7	74.5
4	Token + POS + Drug + Disease	**74.8**	**76.4**
1	Token$^{\mathcal{F}}$	64.2	70.0
4	Token$^{\mathcal{F}}$ + POS + Drug + Disease	71.6	73.8

Figure 5: Multi-input RIMs for SM18-2, personal drug intake. All modules are active

M	Annotations	ELMo	RoBERTa
1	Token	62.6	68.2
2	Token + BirthDef	65.3	70.4
2	Token + Preg	63.8	69.1
2	Token + POS	65.0	69.9
3	Token + POS + BirthDef	67.5	72.2
3	Token + POS + Preg	67.0	71.0
4	Token + POS + BirthDef + Preg	**69.3**	**73.6**
1	Token$^{\mathcal{F}}$	60.3	65.4
4	Token$^{\mathcal{F}}$ + POS + BirthDef + Preg	66.5	69.6

Figure 6: Multi-input RIMs for SM20-5, birth defect in a child. All modules are active

dence for medication intake. Similarly, Figure 6 shows that the *Pregnancy* gazetteer, as a complementary knowledge source, provides effective support for birth defect mention detection.

Some modules active We next evaluate performance when limiting the number of active modules ($m_{Active} < M$). Figures 7-9 show experiments for multi-input RIMs with each annotation as input to different modules. Interestingly, for most tasks, limiting the number of modules yields better performance, corroborating observations made by (Goyal et al., 2019).

This confirms the importance of forcing the annotations into competition mode for the moderate to small datasets: if $m_{Active} < M$, the modules compete for activation. As argued by (Goyal et al., 2019) and (Parascandolo et al., 2018) the competition between modules for representational resources (here the annotations) potentially leads to independence among learned mechanisms, making each module specialize on a simpler sub-problem, which prevents individual RIMs from dominating (Bengio et al., 2020).

Freezing language model vs fine-tuning We are interested in the behaviour of multi-RIMs when the language models are frozen. Freezing models such as BERT has recently demonstrated improvements (including speed-up) in the Adapters framework (Houlsby et al., 2019) and (Pfeiffer et al., 2020). The Adapters rely on injecting new trainable layers (modules) as intermediate layers within a frozen language model. The trainable layers are then expected to learn task specific representations.

Here, we investigate task adaptation using multi-input RIMs, combining trainable modules with complementary task specific resources/representations to compensate for possible losses in learning capacity of the model.

The last two rows in Figures 4-6 report performance when the language model is frozen (no fine-tuning). The fully-featured versions of all frozen systems still outperform the token-only baseline for all tasks for ELMo and almost all tasks for RoBERTa.

All of runs were executed on an Intel® Core i7 2.20GHz CPU. When we fine tune our RoBERTa-based models, the average time for a forward pass and back-propagation for one sample is 1.71sec compared to 0.63sec when the language model is frozen.

This significant reduction in training overhead when freezing language models is helpful for users whose access to computational resources is limited. The reported experiments suggest that appropriate knowledge sources can compensate for losses when freezing heavy language models such as ELMo or RoBERTa.

Comparison with SOTA The SST-2, SE17-4A, SM20-5 tasks have been deployed on GLUE, TweetEval, and Codalab benchmarks respectively, therefore, the state of the art (SOTA) results are available. Current SOTA performances on SST-2 are obtained by (Sun et al., 2019) and (Raffel et al., 2020) (tied), SOTA for SE17-4A is reported by (Barbieri et al., 2020), and SOTA for SM20-5 is reported by (Bai and Zhou, 2020) as shown in Figure 10.

For other tasks however, we replicated the reported SOTA system for each task. For SM18-2 the SOTA performance is reported for (Xherija, 2018), which is a two-layer stacked bi-LSTM with attention. The SOTA results for SE15-11 are reported by *CRNN-RoBERTa* (Potamias et al., 2020) for a RoBERTa-based model in which a bi-LSTM layer is stacked on top of the RoBERTa model, together with a pooling operation for its last layer. The model is replicated here based on hyper-paramters provided in (Potamias et al., 2020).

Figure 10 shows that multi-input RIMs perform at or above SOTA for all benchmarks with greater performance gains for tasks with comparatively smaller datasets and more complex linguistic requirements (SM18-2, SM20-5, SE15-11).

6 Module activation patterns

An advantage of a modular system is the possibility of module inspection. The functionality of each module during the course of processing has to be transparent for assessment.

Figure 11 provides the activation patterns of two multi-input RIMs when applied to two inputs from SST-2 (Figure 11a) and SM20-5 (Figure 11b) to assess whether they give insight into the functionality of the modules.

In Figure 11a, the modules that operate on sentiment knowledge sources (*AFINN*, *MPQA*, and *NRC*) are active only when an annotation is available and are idle (inactive) otherwise. The sentiment modules also compete with one another. Consider *Beautifully* at $t = 1$. For this token, both AFINN and MPQA provide annotations (AFINN:

		SST-2 (Acc %)		SE17-4A (mac-Rec %)		SE15-11 (Cosine)	
M	m_{Active}	ELMo	RoBERTa	ELMo	RoBERTa	ELMo	RoBERTa
5	1	89.6	95.5	65.4	71.0	80.4	82.8
	2	91.7	96.7	67.2	71.9	82.6	84.0
	3	**92.8**	96.9	69.7	**74.5**	84.0	84.8
	4	91.9	**97.5**	**71.3**	73.9	82.9	**85.6**
	5	92.3	97.3	70.4	74.3	**83.2**	85.0

Figure 7: Multi-input RIMs with 5 modules for the sentiment tasks. The number of active modules varies.

M	m_{Active}	ELMo	RoBERTa
4	1	73.1	74.5
	2	75.0	**77.2**
	3	**75.3**	77.0
	4	74.8	76.4

Figure 8: (μF1) of multi-input RIMs with 4 modules (Token + POS + Drug + Disease) on SM18-2. The number of active modules varies.

M	m_{Active}	ELMo	RoBERTa
4	1	68.0	70.4
	2	70.0	73.2
	3	**70.6**	73.3
	4	69.3	**73.6**

Figure 9: (μF1) of multi-input RIMs with 4 modules (Token + POS + BirthDef + Preg) on SM20-5. The number of active modules varies.

+3, MPQA: Pos.), but the AFINN module wins the competition and is active while the MPQA module is inactive. The larger NRC lexicon provides more annotations for the input leading to more activity for the NRC module compared to the other sentiment modules for this sentence.

Inactivity of token modules at certain time steps is particularly interesting, indicating that the model has chosen to attend to a external knowledge source. We find that 63% of the time, when the sentiment lexia provide consistent sentiment polarities, the token module is inactive.

The activation patterns in Figure 11b show the *Birth Defect* and *Pregnancy* gazetteer modules are active only, when an annotation is available. The tokens *CHD* ($t = 9$) and *T18* ($t = 15$) are matched by the *Birth Defect* gazetteer and the token *stillbirth* ($t = 20$) is matched by the *Pregnancy* gazetteer.

The activity patterns are the result of the input selection mechanism (attention). Multi-input RIMs modules are free to select an input signal or ignore it, which allows each module to potentially focus on a specific part of the input. The input selection mechanism prevents the modules from getting updated with spurious inputs (here the input at steps, where no annotation is available). Additionally, this allows the system to develop different modules to select complementary input signals, biasing the behavior away from combining redundant encodings.

We believe that the activation patterns can be useful for model explanation. Nevertheless, the activation patterns have to be studied under a variety of NLP tasks and different, richer annotations, which demands a dedicated study and is beyond the scope of this paper.

Task	SOTA	RIMs
SST-2 (Acc)	**97.5** (1,2)	**97.5**
SE17-4A (mac-Rec)	72.6 (3)	**74.5**
SE15-11 (Cosine)	82.2 (4)	**85.6**
SM18-2 (μF1)	69.2 (5)	**77.2**
SM20-5 (μF1)	69.0 (6)	**73.6**

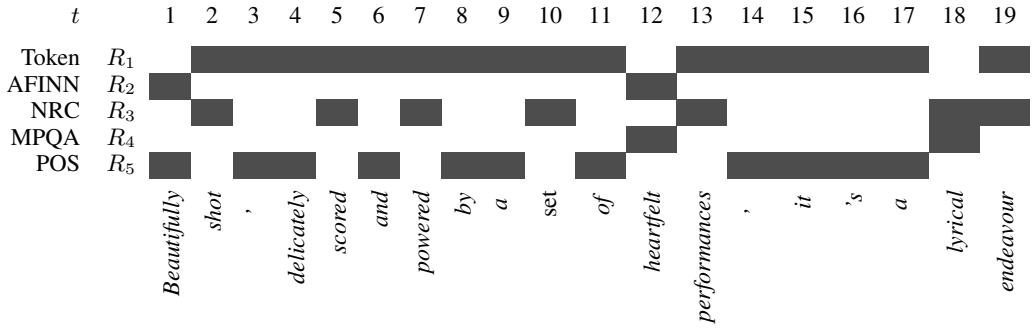
Figure 10: Comparison of the state of the art systems with multi-input RIMs. 1: Ernie (Sun et al., 2019), 2: T5 (Raffel et al., 2020), 3: RoBERTa-RT (Barbieri et al., 2020) 4: CRNN-RoBERTa (Potamias et al., 2020), 5: (Xherija, 2018), 6: (Bai and Zhou, 2020)

7 Conclusion

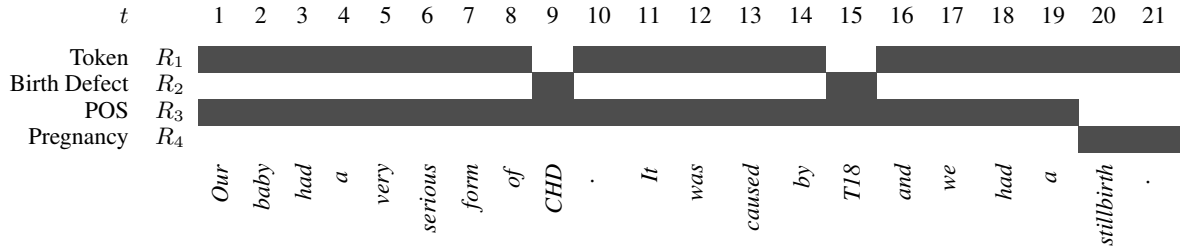
This paper presents proof of concept for a modular system for leveraging different knowledge sources. Under the proposed model, various annotations with different encodings are used as inputs for a set of independent, decoupled, but interacting modules, a novel extension of the RIMs architecture.

Deploying several readily available knowledge sources (gazetteer lists and part-of-speech information), our experiments report on different sentiment tasks and data sets, as well as two health-related tasks and datasets. The results suggest that the modules successfully interoperate for addressing different target tasks and multiple datasets with drastically reduced parameter space (and processing resources).

In addition to the transfer potential of RIMs, we probed their transparency. The activation patterns of the modules in multi-input RIMs showed inter-



a – Input: "*Beautifully shot, delicately scored and powered by a set of heartfelt performances, it's a lyrical endeavour*"



b – Input: "*Our baby had a very serious form of CHD. It was caused by T18 and we had a stillbirth.*"

Figure 11: Activation patterns of the modules of RIMs (ELMo as token embedding) for two samples: (a) SST-2 with $M = 5$ and $m_{Active} = 2$, (b) SM20-5 with $M = 4$ and $m_{Active} = 2$. The gray squares indicate active modules and the white regions indicate inactivity.

estingly differentiated motifs. In particular, the activation patterns show that modules are active only when their input annotation is relevant for the target task. To interpret the functionality of different modules in multi-input RIMs architectures, we plan a detailed analysis of the module activation patterns under different NLP tasks in the future.

References

Parsa Bagherzadeh and Sabine Bergler. 2021. Leveraging knowledge sources for detecting self-reports of particular health issues on social media. In *Proceedings of the 12th International Workshop on Health Text Mining and Information Analysis*, pages 38–48, online.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR'15*.

Yang Bai and Xiaobing Zhou. 2020. Automatic Detecting for Health-related Twitter Data with BioBERT. In *SMM4H 2020*.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650, Online.

Yoshua Bengio. 2017. The consciousness prior. *arXiv preprint arXiv:1709.08568*.

Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Rosemary Ke, Sébastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. 2020. A meta-transfer objective for learning to disentangle causal mechanisms. In *Proceedings of the 8th International Conference on Learning Representations, ICLR'20*.

Kenneth Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. GATE: A Framework and Graphical Development Environment for Robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*.

Aniruddha Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. 2015. SemEval 2015 Task 11: Sentiment analysis of figurative language in Twitter. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 470–478.

Tobias Glasmachers. 2017. Limits of end-to-end learning. In *Asian Conference on Machine Learning*, pages 17–32.

Anirudh Goyal, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf. 2019. Recurrent independent mechanisms. *arXiv preprint arXiv:1909.10893*.

Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, and Yann LeCun. 2017. Tracking the world state with recurrent entity networks. In *Proceedings of the 5th International Conference on Learning Representations, ICLR'17*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR'15*.

Ari Z. Klein, Ivan Flores, Arjun Magge, Anne-Lyse Minard, Karen O'Connor, Abeed Sarker, Elena Tutubalina, Davy Weissenbacher, and Graciela Gonzalez-Hernandez. 2020. Overview of the fifth Social Media Mining for Health Applications (SMM4H) Shared Tasks at COLING 2020. In *Proceedings of the Fifth Social Media Mining for Health Applications (SMM4H) Workshop & Shared Task*.

Shuai Li, Wanqing Li, Chris Cook, Ce Zhu, and Yanbo Gao. 2018. Independently recurrent neural network (IndRNN): Building a longer and deeper RNN. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5457–5466.

Carolyn E Lipscomb. 2000. Medical subject headings (MeSH). *Bulletin of the Medical Library Association*, 88(3).

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Risto Miikkulainen and Michael G Dyer. 1991. Natural language processing with modular PDP networks and distributed lexicon. *Cognitive Science*, 15(3):343–399.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 321–327.

Finn Årup Nielsen. 2011. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *Workshop on 'Making Sense of Microposts: Big things come in small packages'*, pages 93–98.

Canberk Özdemir and Sabine Bergler. 2015a. CLaC-SentiPipe: SemEval2015 Subtasks 10 b,e, and Task 11. In *Proceedings of SemEval 2015 at NAACL/HLT*.

Canberk Özdemir and Sabine Bergler. 2015b. A comparative study of different sentiment lexica for sentiment analysis of tweets. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2015)*.

Giambattista Parascandolo, Niki Kilbertus, Mateo Rojas-Carulla, and Bernhard Schölkopf. 2018. Learning independent causal mechanisms. In *International Conference on Machine Learning (ICML)*, pages 4036–4044.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *Advances in Neural Information Processing Systems 2017*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54, Online.

Rolandos Alexandros Potamias, Georgios Siolas, and Andreas-Georgios Stafylopatis. 2020. A transformer-based approach to irony and sarcasm detection. *Neural Computing and Applications*, pages 1–12.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 Task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. 2012. On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML 2012)*, pages 1255–1262.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. Ernie: Enhanced representation through knowledge integration. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, page 1441–1451.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium.

Davy Weissenbacher, Abeed Sarker, Michael J. Paul, and Graciela Gonzalez-Hernandez. 2018. Overview of the third Social Media Mining for Health (SMM4H) Shared Tasks at EMNLP 2018. In *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop & Shared Task*.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354.

David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. 2018. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 46(D1):D1074–D1082.

Orest Xherija. 2018. Classification of medication-related tweets using stacked bidirectional LSTMs with context-aware attention. In *Proceedings of the 3rd Social Media Mining for Health Applications (SMM4H) Workshop & Shared Task at EMNLP*.