# Opinion Tagging Using Deep Recurrent Nets with GRUs

Alex Adamson and Deger Turan

Stanford University

## Introduction

The fine-grained opinion mining task seeks to, given textual input, find the opinions expressed therein, their intensity and other properties such as their holder and their target. In our work, we focus on the task of tagging attitudes, attitude holders and attitude targets contained in a given sentence. In particular, we first explore the fitness of a deep bidirectional recurrent network for this task using the work described in Irsoy and Cardie 2014 and then attempt to extend it by using gated recurrent units (Chung et al. 2015) instead of vanilla ReLU units.

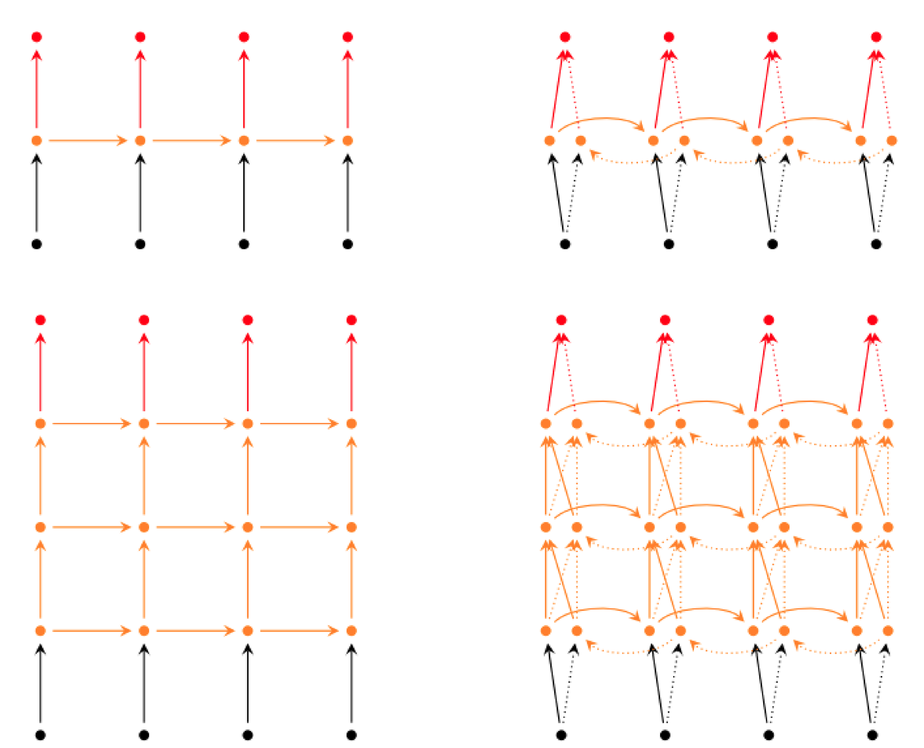

Figure 1: Taken from Irsoy and Cardie 2014: Clockwise from top left: single-layer recurrent neural network, single-layer bidirectional recurrent neural network, deep bidirectional recurrent neural network, deep recurrent neural neural network. Forward layers are denoted with solid lines and backward layers are denoted with dotted lines. Input layers, hidden layers and output layers are shown in black, orange and red respectively.

## Applications

- Refined automated trading (e.g., scale importance of opinion to potential trade based on the credibility of its holder)
- Accelerated parsing of open-ended public polling

## Dataset

We use the MPQA 2.0 dataset (Wiebe, Wilson, and Cardie 2005). The dataset consists of 15737 sentences drawn from the world English-language press. Each sentence is tagged by a trained human for features such as attitudes, attitude targets, attitude holders (agents), objective speech events, and various subjective statements.

### Example annotation

**["The quickest [defeat of Tsvangirai and his MDC lot]$_{\text{target}}$ would come$_{\text{attitude}}$ if they chose the path of violence,"] [an analyst]$_{\text{agent}}$** who spoke on condition of anonymity said.

## Methods

The models we constructed and evaluated include a softmax regression baseline, the deep bidirectional recurrent network described and implemented by Ozan Irsoy (Irsoy and Cardie 2014), and an extension of Irsoy's network that used gated recurrent units instead of ReLU nonlinearities. In particular, whereas Irsoy's model uses the following definition for his interior hidden units

$$\overrightarrow{h}_t^{(i)} = f(\overrightarrow{\underrightarrow{W}}^{(i)}\overrightarrow{h}_t^{(i-1)} + \overrightarrow{\underleftarrow{W}}^{(i)}\overleftarrow{h}_t^{(i-1)} + \overrightarrow{V}^{(i)}\overrightarrow{h}_{t-1}^{(i)} + \overrightarrow{b}^{(i)})$$
$$\overleftarrow{h}_t^{(i)} = f(\overleftarrow{\underrightarrow{W}}^{(i)}\overrightarrow{h}_t^{(i-1)} + \overleftarrow{\underleftarrow{W}}^{(i)}\overleftarrow{h}_t^{(i-1)} + \overleftarrow{V}^{(i)}\overleftarrow{h}_{t+1}^{(i)} + \overleftarrow{b}^{(i)})$$

the GRU model uses

$$\overrightarrow{z}_t^{(i)} = f_2(\overrightarrow{\underrightarrow{Wz}}^{(i)}\overrightarrow{h}_t^{(i-1)} + \overrightarrow{\underleftarrow{Wz}}^{(i)}\overleftarrow{h}_t^{(i-1)} + \overrightarrow{Vz}^{(i)}\overrightarrow{h}_{t-1}^{(i)})$$
$$\overrightarrow{r}_t^{(i)} = f_2(\overrightarrow{\underrightarrow{Wr}}^{(i)}\overrightarrow{h}_t^{(i-1)} + \overrightarrow{\underleftarrow{Wr}}^{(i)}\overleftarrow{h}_t^{(i-1)} + \overrightarrow{Vr}^{(i)}\overrightarrow{h}_{t-1}^{(i)})$$
$$\overrightarrow{\tilde{h}}_t^{(i)} = f(\overrightarrow{\underrightarrow{W}}^{(i)}\overrightarrow{h}_t^{(i-1)} + \overrightarrow{\underleftarrow{W}}^{(i)}\overleftarrow{h}_t^{(i-1)} + \overrightarrow{r}_t^{(i)} \circ \overrightarrow{V}^{(i)}\overrightarrow{h}_{t-1}^{(i)})$$
$$\overrightarrow{h}_t^{(i)} = \overrightarrow{z}_t^{(i)} \circ \overrightarrow{h}_{t-1}^{(i)} + (1-\overrightarrow{z}_t^{(i)}) \circ \overrightarrow{\tilde{h}}_t^{(i)}$$

and analogously for the right-to-left units.

Training the models proved to be difficult. Using a standard supervised error signal and without introducing any weighting to particular kinds of errors, both recurrent networks would converge to something that looked like the prior when trained using minibatch gradient descent.

In order to make the models converge to something meaningful, we took the following steps:

- Introducing a linear transformation of the error signal from the output layer to the error vector passed down by each hidden layer
- Reducing penalties incurred by misclassifying tokens as non-null by multiplying by a constant factor
- Using a momentum rate when performing gradient descent

With these modifications, the models would generally converge without significant oscillatory behavior.

## Results

| | Model | Precision | | Recall | | F1 | |
|---|---|---|---|---|---|---|---|
| | | Prop. | Bin. | Prop. | Bin. | Prop. | Bin. |
| Agent | SR | 0.422 | 0.426 | 0.333 | 0.482 | 0.333 | 0.452 |
| | DRNN | **0.653** | **0.675** | **0.707** | **0.746** | **0.679** | **0.709** |
| | GRU | 0.632 | 0.661 | 0.675 | 0.722 | 0.653 | 0.690 |
| Attitude | SR | **0.339** | 0.341 | 0.192 | 0.544 | 0.245 | 0.419 |
| | DRNN | 0.276 | 0.344 | **0.625** | **0.733** | 0.383 | 0.468 |
| | GRU | 0.291 | **0.360** | 0.575 | 0.672 | **0.386** | **0.469** |
| Target | SR | 0.230 | 0.230 | 0.040 | 0.125 | 0.069 | 0.162 |
| | DRNN | 0.243 | 0.280 | **0.398** | **0.514** | **0.302** | 0.362 |
| | GRU | **0.266** | **0.320** | 0.343 | 0.447 | 0.300 | **0.373** |

Table 1: Results for models from gridsearch over $\lambda$, dropout rate and null-class weight. All models have 25-dimensional hidden units and use the GloVe 300-dimensional Common Crawl embeddings.

We found that introducing the GRUs generally improved performance by a small amount, and that the amount of improvement scaled with the "hardness" of the classification problem, suggesting that the GRU enables the model to learn some abstractions that the vanilla network could not. We had hoped that the GRU network would always outperform the deep RNN considering the models considered when training the GRU include those considered when training the deep RNN (to see this, consider what happens when we set $Wz$ and $Vz$ arbitrarily low and $Wr$ and $Vr$ arbitrarily high).

## References

Chung, Junyoung et al. (2015). "Gated Feedback Recurrent Neural Networks". In: *CoRR* abs/1502.02367. URL: http://arxiv.org/abs/1502.02367.

Irsoy, Ozan and Claire Cardie (2014). "Opinion mining with deep recurrent neural networks". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 720–728.

Wiebe, Janyce, Theresa Wilson, and Claire Cardie (2005). "Annotating expressions of opinions and emotions in language". In: *Language resources and evaluation* 39.2-3, pp. 165–210.