A theoretical framework for Attention

Erik de Godoy Perillo
Advisor: Profa. Dra. Esther Luna Colombini

State University of Campinas

August 21, 2019

1 Introduction

In this document, we briefly formulate a theoretical framework for the concept of Attention. This framework consists of two main parts:

- A *definition* of Attention in terms of its functionalities;
- A *model* of Attention.

Note that the first element aims at answering the question "*What* is Attention?" while the second aims at explaining *how* Attention emerges.

2 A definition of Attention

In this definition, we define a set of *entities of interest* and the phenomenon of Attention in terms of its *functionalities* and how it relates to the entities.

2.1 Why is our definition good?

We believe our definition encompasses what we generally (and intuitively) refer to as attention while being not too broad. Also, the definition is given in terms related to Computer Science so its functionality nicely translates to the domain, which is important since we (so far) intend to develop AI using computers as we know it today. We may not encompass every aspect of Attention and even be conflicting with other definitions. However, this is the set of postulates that we think is the most precise and useful and thus this is what we choose to use for future work to be based on.

2.2 Entities

Below is the list of entities — or "terms" — we use in this work, along with a brief discussion of the meaning we give to each term in the context of this work.

- **Data:** information, stimuli. It may be internal or external. Examples: visual information, audio, memories.
- **Program:** algorithm, sequence of computer (or mental) operations. Programs use data as input in order to carry out a sequence of operations that produces output data and/or actions.
- **Process:** the execution of a program on a specific data instance.
- **Computer:** the executor of processes, the brain.
- **Resource:** when not specified, we mean computational resources, e.g. CPU time.
- **Time:** the flow of time.
- **World:** the external environment.
- **Agent:** the actor in the world.
- **Actions:** the interaction of the agent with the world.
- **Goals:** the ends, objectives to be met.

2.3 What is Attention?

Data, *programs* and *processes* are virtually *infinite*. Computational *resources* and *actions* are finite. *Attention* is *the system of allocating resources to processes*. In other words, *attention* is the entity in *agents* that, given *context* and a set of *processes*, *allocates resources* to execute each of them in order to *produce outputs* in form of *data* and *actions* in a *correct sequential manner* and in *sensible time* in order to reach *goals*.

3 How does Attention happen?

As mentioned in Section 2, in this work we summarize Attention as *the allocation of computing resources to processes*. Thus, when attention is taking place, the system is performing a *selection process*: it is directing its *computing time budget* to certain *programs* which, in turn, are being run with some subset of the possible *data* as input.

In some instances, some of the entities may not be subject to such selection, thus remaining fixed. There may be, for example, systems in which the program to be run and the computing resources are fixed and attention's role is to just select which data to be used. Another example is when both the program and the input data are fixed and Attention selects how much computation time should be dedicated to the process at a given time window.

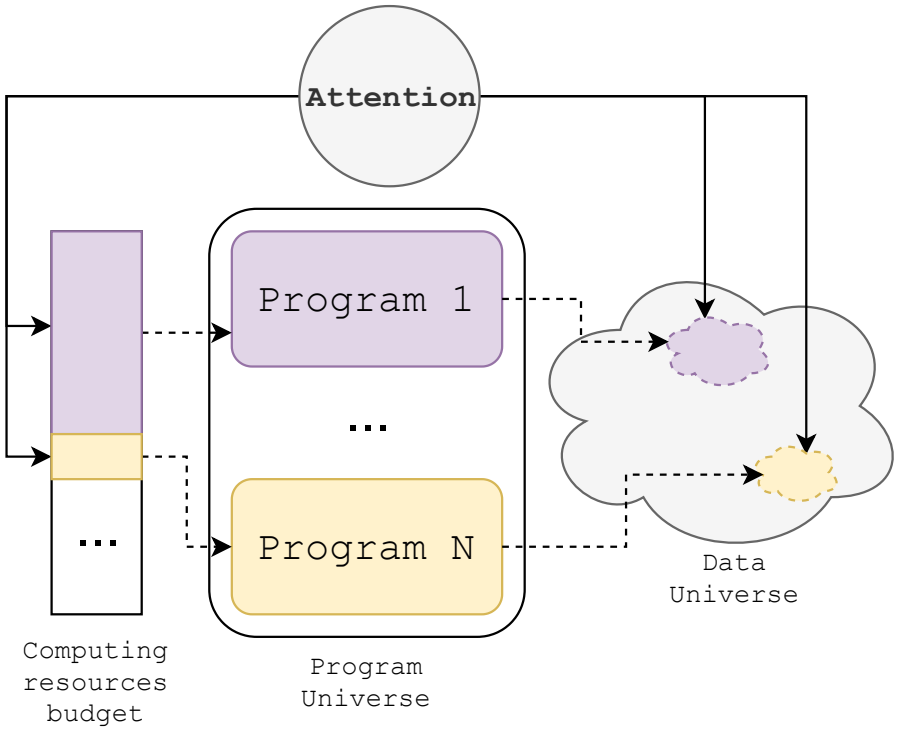

Figure 1: High level overview of the focus phenomenon. Attention is responsible for selecting computation resources for programs and also data to be used by the programs.

The attention process takes place *along the course of time*. At each time step in the process, *inputs* of different classes are used to produce a certain *output*. In a given *timeframe*, the sequence of processing steps produce the emergence of a *selection flow*, or a process. In this section, we approach the entities and taxonomy related to the *end-to-end* process itself. In the next section, we specifically model processing at each time step so that the processes described in this section emerge.

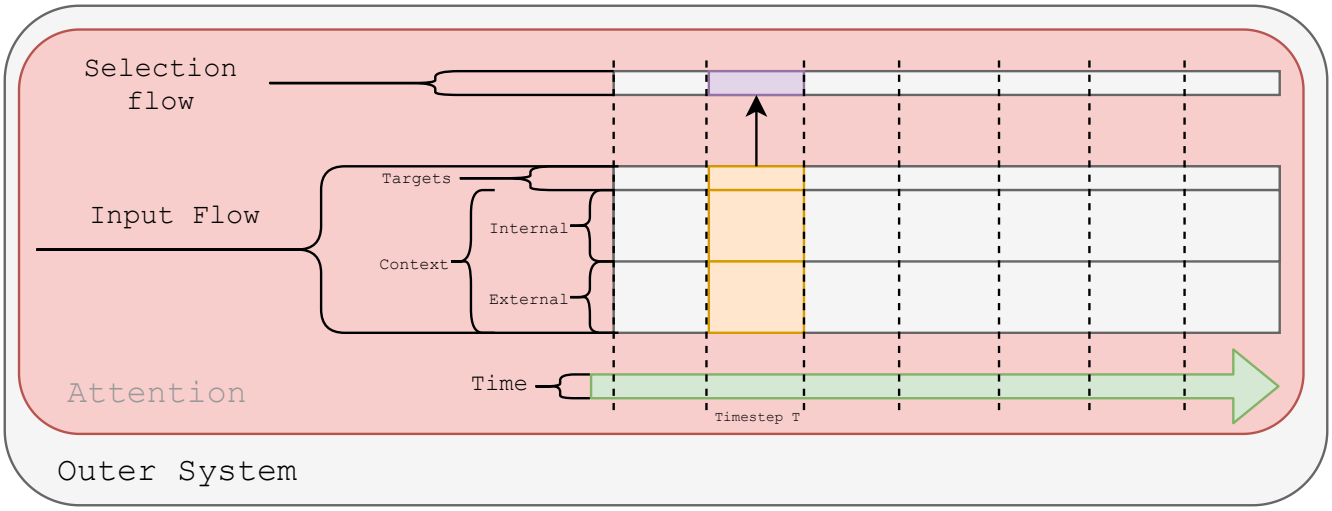

Figure 2: The process of Attention.

3.0.1 Types of Inputs

The information to be used by attention may be classified as either *context* or *targets*. Targets are the subjects of selection by attention. Context is the information used by attention in order to perform the selection. Context can be either *internal* (agent state) or *external* (information about the world) and the context may contain information about the past. An example is an recurrent attentional module for focusing on regions of a video flow. In such case, the hidden state of the network can be considered internal context; the frames can be considered the external context and the pixels of the image are the targets.

3.1 Selection taxonomy

3.1.1 Subjects

The subject of the focus (i.e. the target type) may be 1) *data*, 2) *computing resources* or 3) *programs*.

For example, in a computer vision task of performing classification on an image, the data is the input image, the program is the classification algorithm and the computing resources are the amount of computation (either time or other measure) that will be applied to the task of classifying a subset of the image.

Selection of *data* regards selecting targets that are to be used as input to other processes. We can further classify this selection based on the nature of the data being selected, such as:

- **Feature-based**: based on the features of a stimuli, such as color, orientation
- **Location-based**: based on the location of the stimuli, such as the coordinates of a pixel.

Selection of data is the most common of selection nowadays in Deep Learning systems. An example of a process of selection of data is the selection of a window of the input image to be processed in an image classification task. In this case, selection can be considered to be location-based.

Selection of *computing resources* is the selection of the amount of computation to be used for a given process One example is a recurrent neural network that performs substeps at each time step and at each step decides how many substeps to perform.

3.1.2 Continuity

The continuity of the process may be even 1) *hard* or 2) *soft*. This division has been popular in Deep Learning research lately.

For *hard* attention, the selection is discrete: the process performs selection of a subset of the targets. In the case of data, one example of hard selection is the selection of a specific subset of k feature vectors (from $m \geq k$ options) to be further processed.

For *soft* attention, the selection is continuously spread accross the targets. It can be seen as of type of 'weighting" spread accross the targets. Using the example above, the selection of feature vectors could be soft — in this case, instead of selecting a subset of vectors, every vector could be given a weight $0 \leq w \leq 1$ for a further convex combination of the vectors.

3.1.3 Flow

The flow of selection along time in a process can be classified as 1) *Ephemeral* or 2) *Enduring*.

Ephemeral refers to selection of subjects in the context of a short time window. For example, in a task of visual control of a car, an attention process to identify the abrupt appearance of moving obstacles could be considered ephemeral since it does not take into account much context. On the other hand, an attentional process that keeps focus on a part of the road is not ephemeral, since it's long-termed.

Enduring refers to selection of objects on a long time window. Using the example above, focusing on the road would be considered an enduring process. The enduring focus can be further classified as:

- **Oriented**: An arbitrary focus sequence so as to complete a certain task.
- **Sustained**: Focus restrained to a subset of the targets.
- **Divided**: Focus alternating among a subset of the targets.

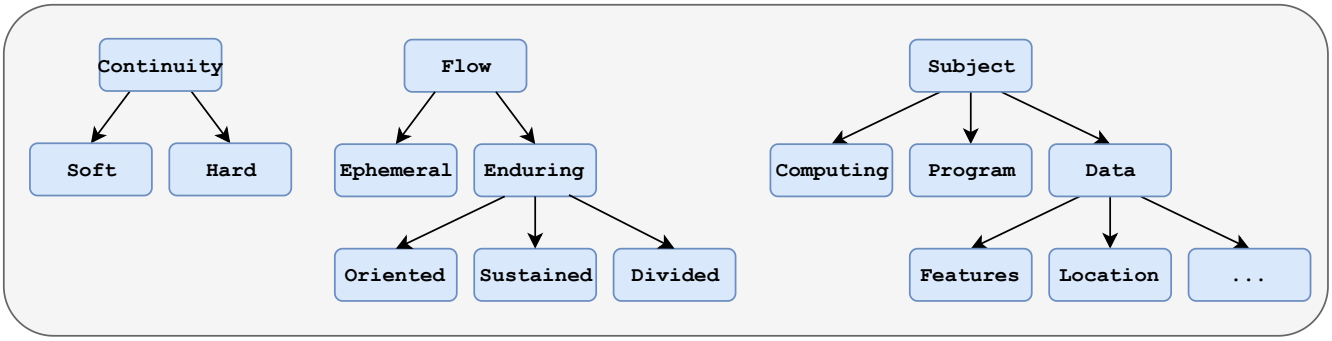

Figure 3: Attention attributes.

4 Attentional Module

We propose that the process of Attention — as discussed in Section 3 — can emerge by means of *a series of components* — which we call *attentional modules*. These modules can alter data being processed and the execution flow of the algorithm and provide the functionalities of Attention.

Figure 4: Attentional module.

Figure 4 illustrates the attentional module. At each time step t, the module receives as *input*:

- Current *outer state* $o_t \in O$, where O is the *outer state set*.
- Group of *focus targets* $\tau_t = \{\tau_{t1}, \ldots, \tau_{tk}\}, \tau_{ti} \in T$, where T is the *focus target set*.
- Past *inner state* $\iota_{t-1} \in I$, where I is the *inner state set*.

The module produces as *output* (as a function of both inputs):

- Current *inner state* $\iota_t \in I$.
- Current *focus output* $\alpha_t = \{\alpha_{t1}, \ldots, \alpha_{tk}\}, \alpha_{ti} \in A$, where A is the *focus output set*.

The focus output is the main element of the module: it can be used to allocate *finite resources* to a set of "candidate targets" by giving them an "importance score" which can be used in any arbitrary way in following steps. Each element α_{tk} is respective to a target element τ_{tk}.

4.1 Modules forming an attentional system

A system with Attention may contain more than one attentional modules — even in a recursive manner. Together, these modules always perform the function to provide focus as discussed in Section 3. We now provide some examples of how the attentional modules can act to provide such functionalities.

4.1.1 Soft and Hard focus

The different focus continuities can be obtained by means of attentional modules in the following manner:

- **Soft Focus**: $A = [0, 1]$, with $0 \leq \sum_{i=1}^{k} \alpha_{ti} \leq 1$
- **Hard Focus**: $A = \{0, 1\}$, with $0 \leq \sum_{i=1}^{k} \alpha_{ti} \leq M$ and $0 \leq M \leq |\tau_t|$

4.1.2 Example of an entire system

Figure 4.1.2 shows the diagram of a possible system with attention. The module *TaskATT* uses hard attention to select a certain task k to be executed for some time at time step t. Among the computations of task k, there is the module *DataATT*, which uses soft attention to allocate resources to a set of items. It is worth noting that time is relative to each attentional module: *TaskATT* has a temporal course over time steps t that is different from that of *DataATT*, which is over time steps t'. Also, their sets of inputs and outputs may differ. Together, these modules provide two types of focus: *hard*, *enduring/oriented* focus on *programs* and *soft*, *ephemeral* focus on data.

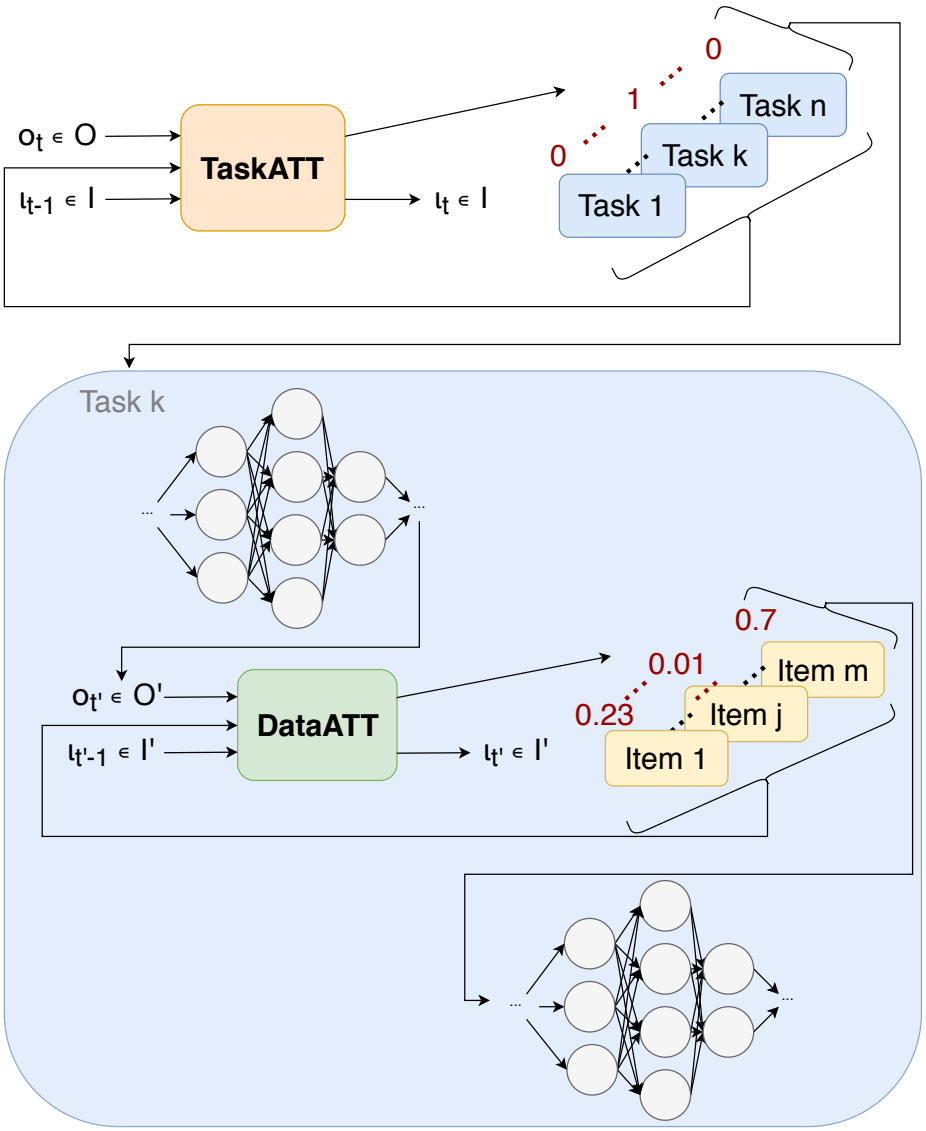

Figure 5: Example of a system that uses attention.

5 Validating the framework

In this section, we investigate some recent works on attention and see how they fit in our framework.

5.1 Image Caption Generation

The work [4] is among the first to propose using attention to image caption generation: the encoding of the input image is represented as a set of vectors — each respective to a certain spatial region of the image — and the attentional component gives weights to each vector at each step in order to produce another vector to be used in further computations.

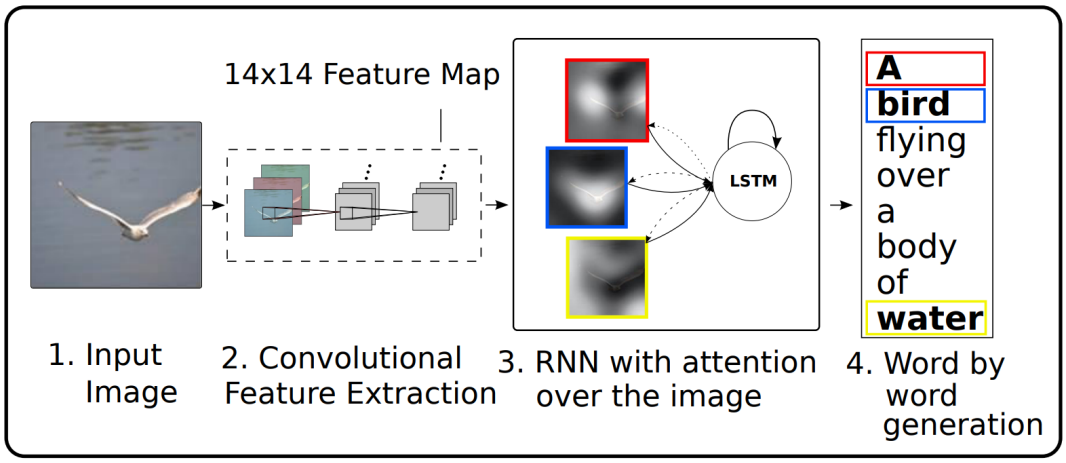

Figure 6: Diagram of natural language image description using Attention (from [1]).

While there are many abstract processing substeps in the process, the *end-to-end* effect is that of a focus with *soft* continuity, with *enduring* and *orienting* flow, targetting *data* on a *Location-based* manner. Figure 5.1 illustrates the proposed model.

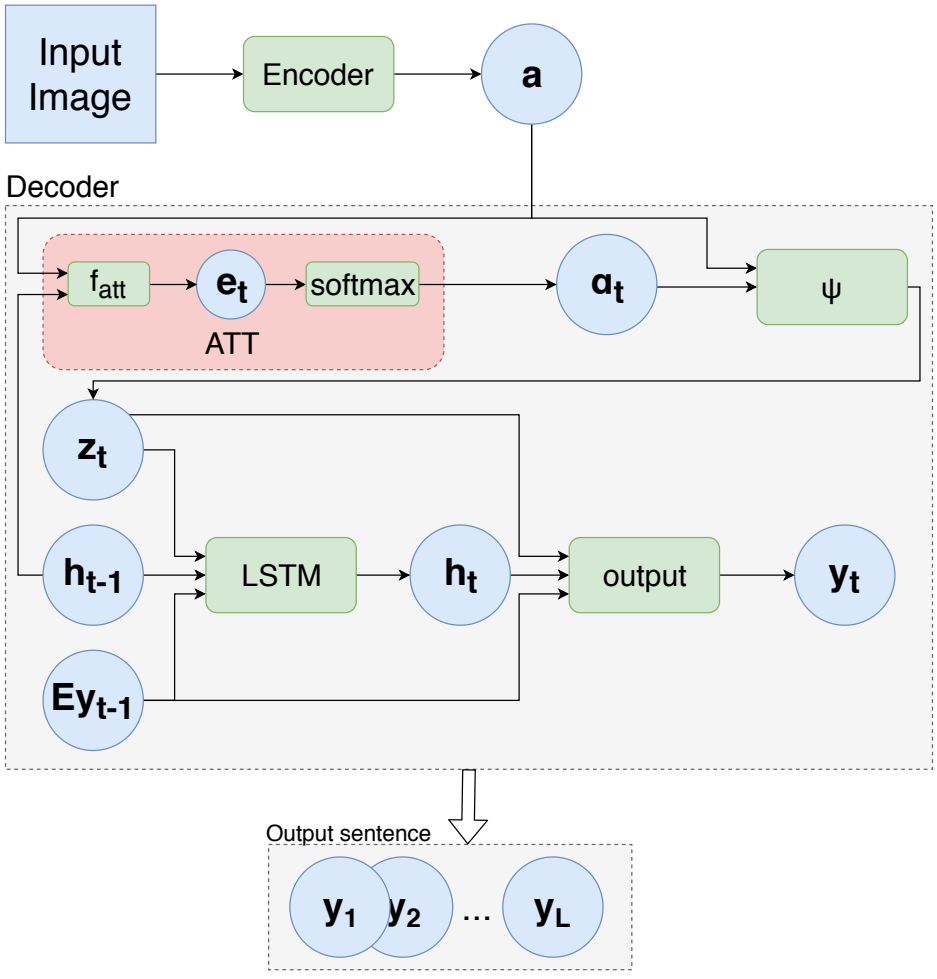

Figure 7: Model proposed for image captioning in [4] with attentional module.

Figure 5.1 illustrates the model proposed in the work. The steps that calculate the attention to each

encoding vector can be "encapsulated" as an attentional module under our modeling: a, the input image encoding, is the *focus target input* τ_t, h_{t-1}, the hidden state of the model LSTM, is the *outer state input* o_t and α_t, the weights given to each encoding vector, is the *focus output*. In this case, $A = [0, 1]$. Note that, in this case, the *internal state* is empty.

5.2 Adaptive Computation Time

The work [2] proposes an RNN that can perform a variable number of computation "sub-steps" for each time step t'. The main idea is to calculate an amount $0 \leq p_{t',t} \leq 1$ to be "spend" for each computation sub-step t up until the moment the total spent reaches the "budget" of 1 (in which moment the computation is halted). The final value $y_{t'}$ is computed as an weighted average of the intermediate $y_{t',t}$ values and the weights are the values $p_{t',t}$.

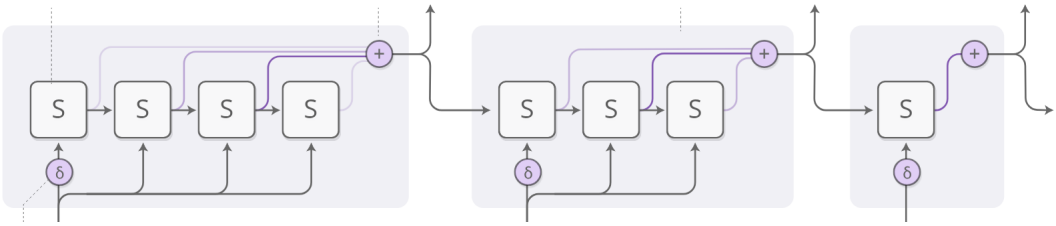

Figure 8: Adaptive computation time process illustration.

The attention component provides two types of focus: The selection of computing substeps at each step can be thought as a focus flow with *hard* continuity, *enduring/oriented* signature and *computing* as focus target. The computation of the result of each step uses *soft* focus of *ephemeral* signature and *data* as focus target.

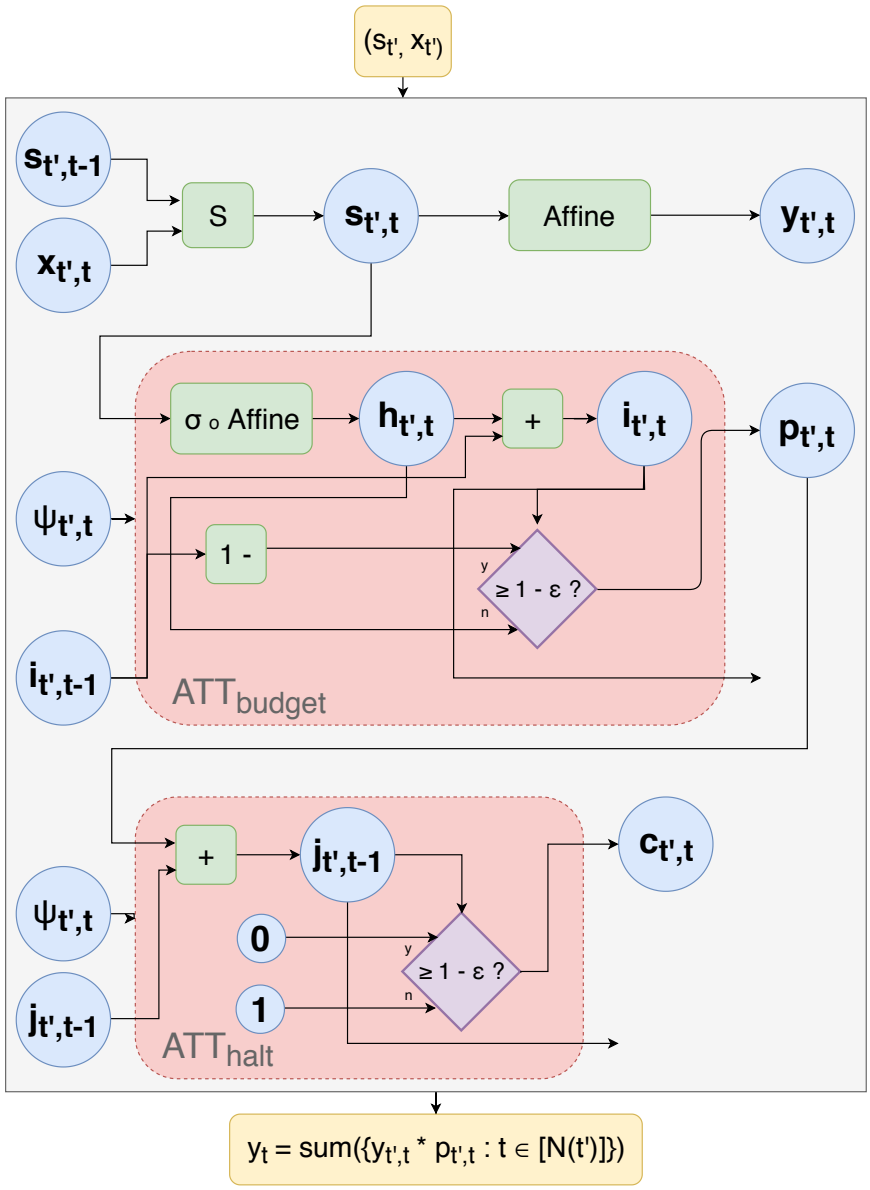

Figure 9: Model proposed for image captioning in [2] with attentional module.

Figure 5.2 illustrates the model proposed in the work. The proposed model can be thought as having two attention modules:

- ATT_{budget}, which computes the value $0 \leq p_{t',t} \leq 1$ to be spent at a given sub-step. In this analogy, $s_{t',t}$ — the state of the RNN cell — is the *outer state* o_t; ψ_t — a dummy element representing the current computation sub-step — is the *target* τ_t; and $i_{t',t}$ is the *inner state*. The *focus output* $p_{t',t}$, besides representing values to be consumed from the budget, can be thought of as an importance weight for the final output y_t, since the produced values are used to computed the weighted average.

- ATT_{halt}, which computes the value $c_{t',t} \in \{0, 1\}$, which is 1 if the cell should continue further sub-steps and 0 otherwise. In this analogy, $p_{t',t}$ is the *outer state* o_t; ψ_t — a dummy element representing the current computation sub-step — is the *target* τ_t; and $j_{t',t}$ is the *inner state*.

5.3 Recurrent Attention Model of Visual Attention

The work [3] proposes a general recurrent model that uses visual attention at each step by selecting a "retina-like" representation of a portion of the input image to carry out further computations. At each time step t, the model uses the selected location l_{t-1} to extract a retina-like representation from input image. An arbitrary action a_t can be executed to possibly alter the environment.

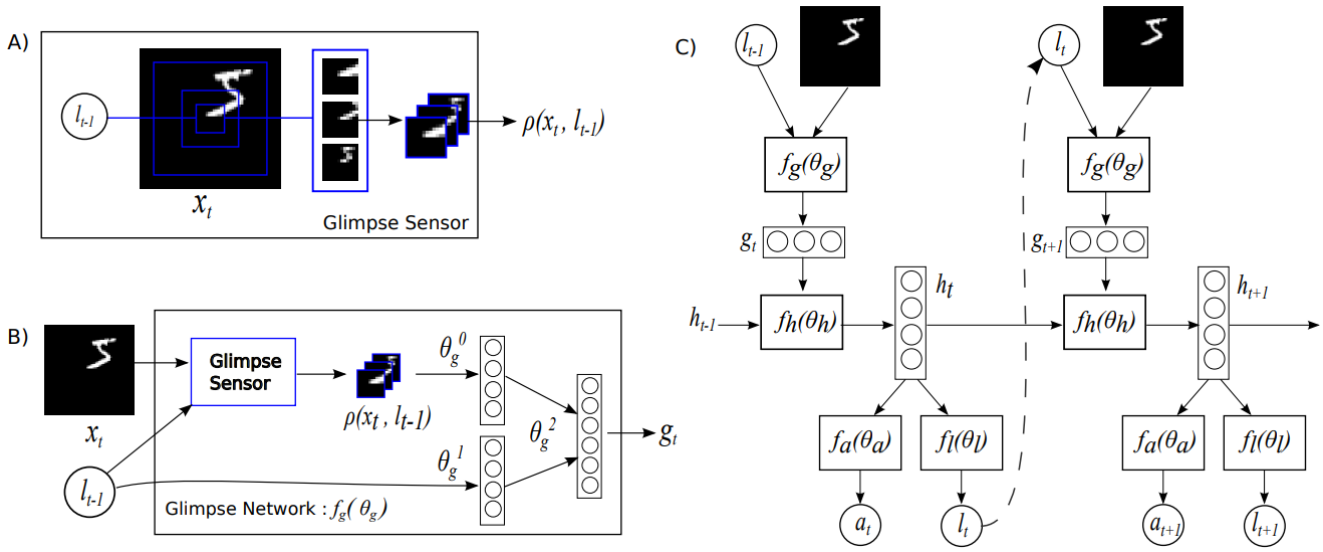

Figure 10: General recurrent architecture proposed in [3].

The attention component in the proposed model can be thought as providing *hard*, *enduring/oriented* focus to *data* in a *location-based* manner.

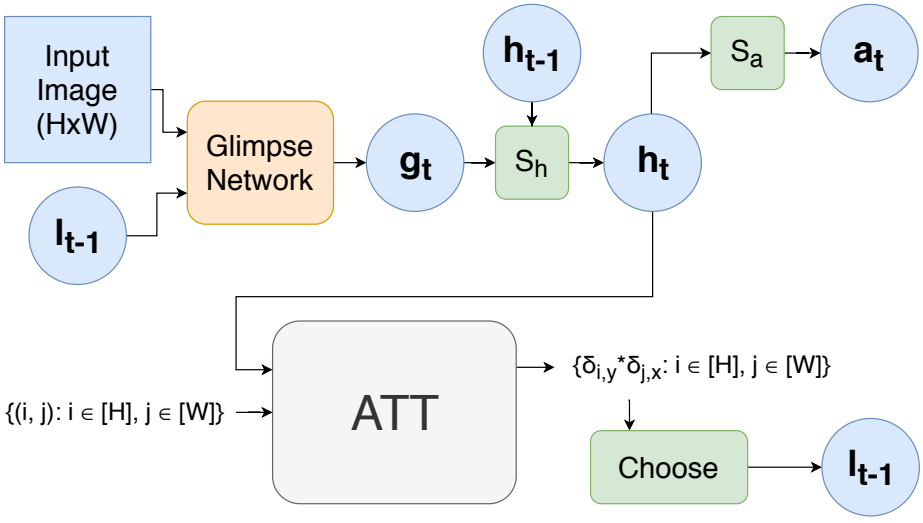

Figure 11: General recurrent architecture proposed in [3] with attentional module.

Figure 5.3 illustrates the model proposed in the work. In this representation, the hidden state of the RNN h_t is the *outer state* input o_t; The set of possible pixel coordinates $\{(i,j) : i \in [H], j \in [W]\}$ (with H, W as the height, width of the image) is the *focus targets* input τ_t; and the set $\{\delta_{i,y}\delta_{j,x} : i \in [H], j \in [W]\}$ is the *focus output*. Note that only the element $\delta_{i,y}\delta_{j,x}$ — which is respective to the chosen pixel coordinates (x, y) is equal to 1.

Table 1 summarizes the taxonomy of the works cited above.

Table 1: Taxonomy of cited works.

Work	Focus continuity	Focus flow	Focus subject
[4]	Soft	Enduring/Oriented	Data
[2]	Hard	Enduring/Oriented	Computation
[2]	Soft	Ephemeral	Data
[3]	Hard	Enduring/Oriented	Data/Location-based

References

[1] KyungHyun Cho, Aaron C. Courville, and Yoshua Bengio. "Describing Multimedia Content using Attention-based Encoder-Decoder Networks". In: *CoRR* abs/1507.01053 (2015). arXiv: 1507.01053. URL: http://arxiv.org/abs/1507.01053.

[2] Alex Graves. "Adaptive Computation Time for Recurrent Neural Networks". en. In: *arXiv:1603.08983 [cs]* (Mar. 2016). arXiv: 1603.08983. URL: http://arxiv.org/abs/1603.08983 (visited on 09/11/2018).

[3] Volodymyr Mnih et al. "Recurrent Models of Visual Attention". In: *arXiv:1406.6247 [cs, stat]* (June 24, 2014). arXiv: 1406.6247. URL: http://arxiv.org/abs/1406.6247 (visited on 09/11/2018).

[4] Kelvin Xu et al. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention". In: *CoRR* (2015). URL: http://arxiv.org/abs/1502.03044.