Summer_Paper_Reading_2016

eg: 0-ACL16-Ng-Paper_title.pdf

- Abstract
  - Overview:
  - Advantage:
  - Disadvantage:
  - What can I do? / Can I employ its idea?
- Experiments
  - DataSet:
  - Toolkit:
  - Baseline:
  - Result:

[TOC]

1-NIPS15-Google-Pointer Network

PDF, Bib

- [**Problem-Paper**] the number of target classes depends on the length of input, which is variable.
- [**Problem-Experiment**]
  - finding planar convex hulls
  - computing Delaunary triangulations
  - the planar Travelling Salesman Problem
- [**Model**] instead of using attention to blend hidden units of an encoder to a context vector of a decoder, it uses attention as a pointer to select a member of the input sequence as the output.
- [**Remark**]
  - Neural attention model deals with the fundamental problem representing variable length dictionaries, appling to three distinct algorithmic problems.
  - Idea is fantastic.

Authors of this paper propose an attention model based on "Neural machine translation by jointly learning to align and translate"[1] by using LSTMs. Their model decides what to predict by using the output of a softmax MLP over the inputs as opposed to regular attention model used in [1](RNNSearch), where they used that to use for the convex combination of the input annotations to predict where to attend to translate in the target language. The output of the softmax-MLP predicts the location of the output at each timestep directly from the input sequence.

The advantage of this approach is that, the output softmax dictionary only depends on the length of the input sequence instead of on the number of possible instances in the problem. The authors have provided very interesting experimental results on three different discrete optimization problems.

2-ACL16-IBM-Addressing Limited Data for Textual Entailment Across Domains

PDF, !Bib

- [**Problem-Paper**] exploit unlabled data to improve F-score for TE task.
- [**Problem-Experiment**] find all sentences in a corpus that entail a given hypothsis.
  Domain: Newswire (RTE-5,6,7) & Clinical (self construct)
- [**Model**] Tradition Features + self-training/activate learning
- [**Conclusion**] The author analysis the possible explanation for the improvement: Class Distribution. down-sampling and up-sampling is not useful. Activate learning will sampling positive examples, thus it can match the performance with fewer examples.
- [**Remark**] Experiment is beautiful and convising, Information Retrival Method.

This paper illustrates that how self-training will influence the classification and why active learning will reduce the examples to only 6 percent.
The author experiments on two domains -- newswire and clinical. First, the author creates an entailment dataset for clinical domain with human annoted. Second, he builds a highly competitive supervised system, called ENT. Last, he explore two strategies - self-training and active learning to address the lack of labled data. Experiment is done in detail and convincing.

3-ACL16-Stanford-A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task

PDF, !Bib, !Github

- [**Problem**] CNN/Daily Mail Reading Comprehension Task
- [**Model**]
  - Traditional Features
    - Feature ablation analysis
  - Attention Neural Network(followed 5-NIPS15-NYU-End-To-End Memory Networks)
- [**Related Dataset**]
  - CNN/Daily Mail (2015)
  - MCTest（2013）
  - Children Book Test （2016）
  - bAbI （2016）
- [**Data Analysis**] breakdown of the Examples, Analysis the perfomance on each categories(although on small dataset).
- [**Remark**] Also we can construct traditional ML and NN, **data analysis is important**, without this, Experiment seems to be inconvincing.

This paper conducts a thorougn examination of CNN/Daily Mail Reading Comprehension Task, which origin from the idea that a bullet point usually summaries one or several aspects of the article. **If the computer understands the content of the artticle. It should be able to infer the missing entity in the bullet point.**

two supervised systems are implemented -- a conventional entity centric classfier and an end to end neural network. Expriment shows that the straight-forward NLP system, compared with origin frame-semantic parser[^1], obtain accuracies of 72.4% and 75.8％ on these two dataset ｓ.

Besides, the author **extracts 100 examples to analysis the results**. She roughly classify the examples into 6 categories, i.e., Exact Match, Paraphrase, Parial clue, Multiple sent, (Coref.Error, Hard), the last two is hard for human to obtain the correct answer.

[^1]: Teaching Machine to read and comprehend, NIPS15, Hermann et.al

4-NAACL16-CMU-Hierarchical Attention Networks for Document Classification

PDF, Bib

- [**Goal**] Hierarchical Attention Networks for Document Classification
- [**Problem**]
  - a). Sentiment Estimation
    - Data Set: Yelp reviews, IMDB reviews, Amazon reviews
  - b). Topic Classification
    - Data Set: Yahoo answers
- [**Model**]
  - (i) it has a hierarchical structure that mirrors the hierarchical structure of documents;
  - (ii) it has two levels of attention mechanisms applied at the word and sentence-level, enabling it to attend differentially to more and less important content when constructing the document representation
  - The context vector u_w can be seen as a high level representation of a fixed query “what is the informative word” over the words like that used in memory networks (Sukhbaatar et al., 2015; Kumar et al., 2015).
- [**Remark**]
  - **Modification of Model**: Hierarchical + Attention

5-NIPS15-NYU-End-To-End Memory Networks

PDF, Bib, Theano, Tensorflow

- [**Goal**] introdce a **neural network** with a **recurrent attention** over a possibly large **external memory**.
- [**Problem-Experiment**]
  - a). synthetic question answering (Reading Comprehension Task)
  - b). Language Model
- [**Dataset**]
  - a).bAbI
  - b).Penn Tree Bank & Text8

- [**Model**]
  Notation:
  - input: sentences: $$x*1, x*2, \ldot, x_i$$, question: q
  - output: answer:a
  - Variable: A, B, C, W
  - $$ shape(x*i) = d; shape(A) = (d, V); shape(m*i) = d; shape(input) = (len(sentences), d) $$
  - Sentence Representation: BoW & **Position Encoding** (slightly different from BoW, add position information), $$ m_i = \sum*{j}Ax*{ij} $$
  - Temporal Encoding: QA tasks require some notion of temporal context, $$ m*i = \sum*{j}Ax*{ij} + T*{A}(i)$$
  - Random Noise: to reqularize T_A, randomly add 10% of empty memories to the stories.
  - The capacity of memory is restricted to the most recent 50 sentences.
  - **Since the number of sentences and the number of words per sentence varied between problems, a null symbol(all zero) was used to pad them all to a fixed size**
- [**Remarks**]
  - **How to write a new Model with not the state-of-art performance**?
    - incuction previous model to this model (LSTM, Attention, ...)
    - Compare with other related model (where is the difference?)
    - How can the model changes?
    - What can the model apply?
  - **Related Works** deserves to be learned.
  - **Code** deserves to be implenment by myself.

This Paper introduce a neural nerwork with a recurrent attention model over a possibly large external memory.

The memory in RNNs model is the state of the network, which is latent and inherently unstable over long timescales. The LSTM-based Model address this through local memory cells which lock in the network from the past. This model differs from these in that it uses a global memory, with shared read and write functions.

This model also related to attention mechanism in Banhadnau's work[^1], although Banhadnau's work[^1] is only over a single sentence rather than many.

This approach is compettive with Memory Networks, but with less supervision.

Next is quoted from yanranli's blog. The summary deserves me to leaned. you can refer to here for more details.

> And the authors attempt several ways in this paper to fulfill their goal. First, the single-layer or multi-layer, and then the transformation of feature space. If one separate the output of the end-to-end memory networks, they can be parallized with typical RNN. The output comprises of two parts, internal ouptut and external output, which can be parallized to RNN's memory and predicted label, respectively.

[^1]: Neural machine translation by jointly learning to align and translate. ICLR15, Bahdanau, Cho, and Bengio

6-NAACL16-Sultan-Bayesian Supervised Domain Adaptation for Short Text Similarity

PDF, Bib

- [**Problem**] Domain Adaptation for Short Text Similarity
- [**Model**] A two-level hierachical Bayesian model -- Each w_d depends not on its domain-specific observations (first level) but also on information derived from the global, shared parameter $w*$ (second level). And the hierarchical structure (1) jointly learns global, task-level and domain-level feature weights, (2) retaining the distinction between in-domain and out-of-domain annotations.
- [**Features**]
  - monolingual word aligner
  - cosine similarity from 400-dimensional embedding(Baroni et.al, 2014)
- [**Experiment**]
  - a). Short Text Similarity(STS), 10 domains
  - b). Short Answer Scoring(SAS), Dataset: Mohler et al., 2011
  - c). Answer Sentence Ranking(ASR), Dataset: Wang et al., 2007, TREC8-13
- [**Remarks**]
  - Although this is traditional feature method, and the results is not inspiring, the author construt **amount of Analysis** to show the advantage of the system and answer why it does not perform well(**because of the data, smile**).

7-NAACL16-Lu Wang-Neural Network-Based Abstract Generation for Opinions and Arguments

PDF, Bib

Excellent work, clear structure, read it more times

- [**Problem**] Abstract generation for opinioons and arguments
- [**Model Step-by-step**]
  - Data Collection, the dataset can be found here:
    - movie reviews, from www.rottentomatoes.com
    - arguments on contraoversial topics, from idebate.org
  - Step 1: **Problem Formulation**, the ... task is defined to as finding y, which is the most likely sequence of word... such that: formulation
  - Step 2: **Decoder**, LSTM model for long range dependencies.
  - Step 3: **Encoder**, Bi-LSTM + Attention, Attention is userd to know how likely the input word is to be used to generate the next word in summary.
  - Step 4: **Attention Over Multiple Inputs**: It depends on task.
  - Step 5: **Importance Estimation to sub-sampling from the input**: because there are two problems with this approach. Firstly, the model is sensitive to the order of text units (a paragraph); Secondly, time cost too much.
  - Step 6: **Post-processing**: re-rank the n-best summaries; it is directly related to the final goal.
- [**Experiment:anwser the question from model**]
  - Question 1: How is the performance of component? -- Importance Estimation Evaluation(Step5)
  - Question 2: What is the model performance for automatic summary?
  - Question 3: What is the model performance according to human?
  - Question 4: What is the hyper-parameter K in sub-sampling effect? (Step5)
  - Question 5: Is the post-processing needed? (Step6)

This work comes from deepmind, it presented a neural approach to generate abstractive summaries for opinionated text. Attention-based method is employed to find salient inormation from different input text generate an informative and concise summary. To cope with the large number of input text, an importance-based sampling mechanism is deployed for training.

This work applies the attention model to abstract generation. **I think the motivation to build this model is to employ attention over different input text (different task may have different question to solve, different model to modify, haha)**

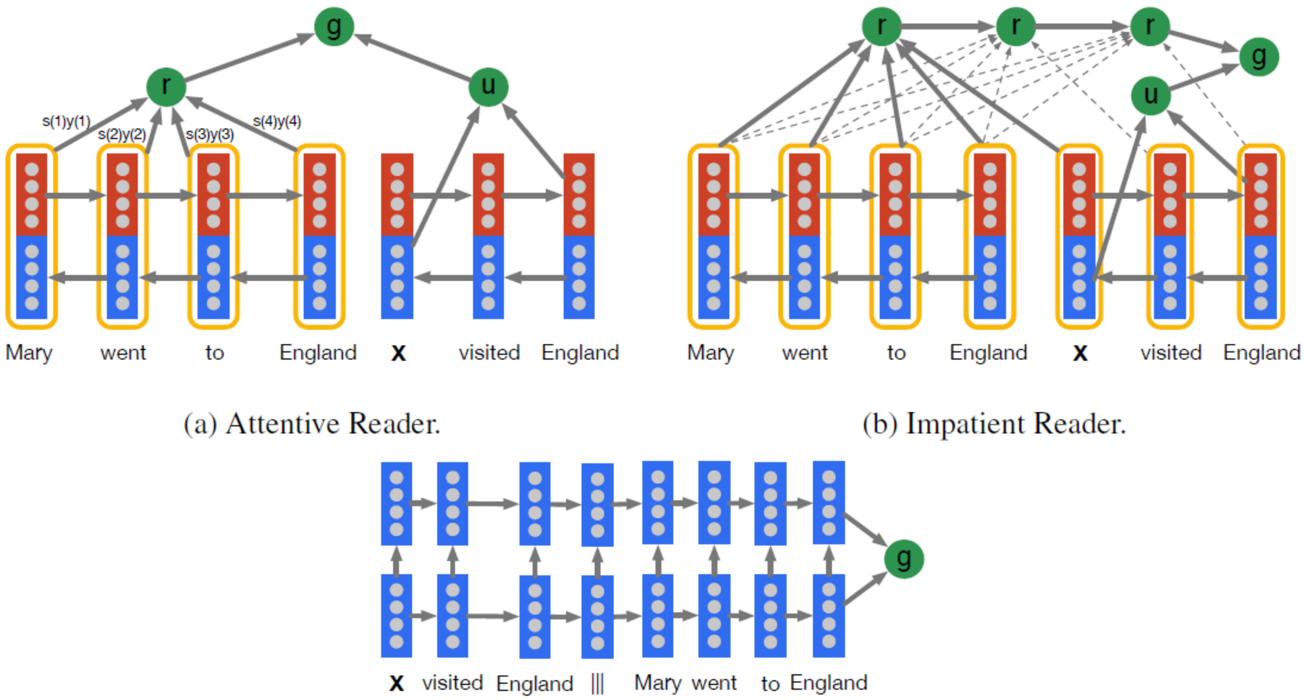
I like this writting skills and research method, such as Bengio, deepmind.

8-NIPS15-DeepMind-Teaching Machines to Read and Comprehend

PDF, Bib, Tensorflow

What is the guide paper, which will be reseach hot in the future?

- [**Problem**]: Large scale training and test datasets have been missing for Reading Comprehension Task (Document, Question) - (Answer).
- [**Data Collection**]: Summary and paraphrase sentences, with their assosiated documents, can be readily converted to context-query-answer triples using simple entity detection and anonymisation algorithms.
- [**Models**]
  - Traditional Methods(Benchmark): Symbolic Matching Models.
  - Neural Network Models



(a) Attentive Reader.

(b) Impatient Reader.

(c) A two layer Deep LSTM Reader with the question encoded before the document.

Figure 1: Document and query embedding models.

- g(d, q) returns a vector embedding of a document and question pair
- **The Attentive Reader can be viewed as a generalisation of the application of Memory Networks**, Memory Networks employs an attention mechanism at the sentence level, while this at the tokens level but with entire future and past context.
- **Impatient Reader emphasizes on the reread mechanism.** Motivated from " We expect to comprehend deeplier as we reread once more "

> Impatient Reader. This model is really interesting and intuitive, which emphasizes on the reread mechanism. That is to say, for each token in each query, a whole document is read once through. One token, one document. Next token, again this document. That's what the reread is. I interpret such reread mechanism as a gradual comprehension procedure. When a human being is reading a tough article, s/he will read again and again. We expect to comprehend deeplier as we reread once more. Such motivation, behind this reread mechanism, if any, will make larger impact when predicting beyond a token level output. Therefore, I think this mechanism is worthy of implementing in more tasks.

- [**Experiments**]
    - How difficult is this task? Traditional excellet model, **Danqi Chen [^1] achieve great results with Features**
    - Traditional Methods versus Neural Models?
    - Which component contributes to the end performance?

[^1] 3-ACL16-Stanford-A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task

9-AAAI16-Mueller-Siamese Recurrent Architectures for Learning Sentence Similarity

PDF, Bib

A new top resluts for Relateness on SICK: 0.8822(pearson)

- [**Model**]
    - replace the top softmax layer with l1 norm similarity function
    - Sentence Representation can capature the following information:
        - negation, not, no
        - topic
        - Entailment classsification
    - I think the model is to enforcement the similarity of the two sentences, which is direct and efficient.

- [**Experiments TODO**]
    - replace the top layer
    - learn how to fine-grain the network.
    - Read this paper again.
    - Read the latest related paper from reference.

10-NIPS15-Ryan-Skip-Thought Vectors

PDF, Bib, Review Theono

This paper presented an encoder-decoder model to learn a generic, distributed sentence encoder.

- [**Problem**]: Sentence Representation. Sentence vectors ate extracted and evaluated with linear models on 8 tasks.
- [**Motivation**]:

> In this paper we abstract away from the composition methods themselves and consider an alternative loss function that can be applied with any composition operator. **(step1 purpose: what we want to do?)**
>
> We consider the following question: is there a task and a corresponding loss that will allow us to learn highly generic sentence representations? **(step2 question: The key problem?)**
>
> We give evidence for this by proposing a model for learning high-quality sentence vectors without a particular supervised task in mind. **(step3 plan1: solution we reject)**
>
> Using word vector learning as inspiration, we propose an objective function that abstracts the skip-gram model to the sentence level. That is, instead of using a word to predict its surrounding context, we instead encode a sentence to predict the sentences around it. Thus, any composition operator can be substituted as a sentence encoder and only the objective function becomes modified. **(step4 our work: step by step, more and more deep)**
>
> Figure 1 illustrate the model. We call our model **skip-thoughts** and vectors induced by our model are called **skip-thought vectors**. **(step5 Others: Figure or Name)**

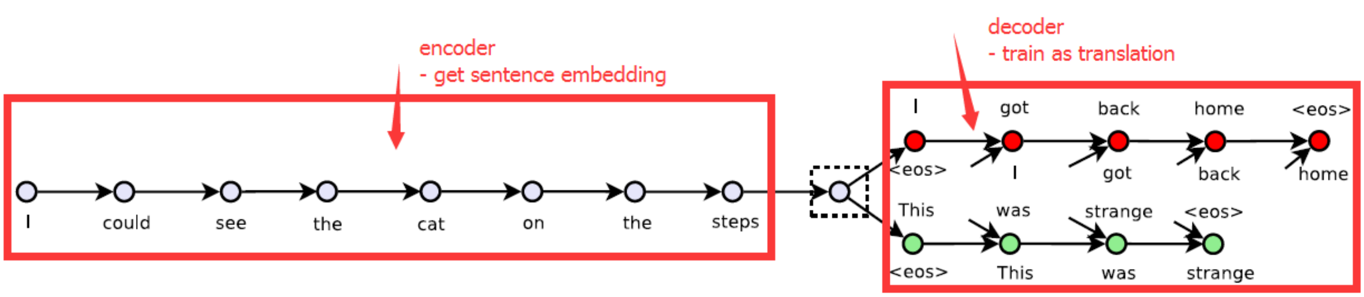


Figure 1: The skip-thoughts model. Given a tuple (s_{i-1}, s_i, s_{i+1}) of contiguous sentences, with s_i the i-th sentence of a book, the sentence s_i is encoded and tries to reconstruct the previous sentence s_{i-1} and next sentence s_{i+1}. In this example, the input is the sentence triplet *I got back home. I could see the cat on the steps. This was strange.* Unattached arrows are connected to the encoder output. Colors indicate which components share parameters. $\langle \text{eos} \rangle$ is the end of sentence token.

11-EMNLP15-Hua He-Multi-Perspective Sentence Similarity Modeling with Convolutional Neural Networks

PDF, Bib

Multi-Perspective: 1). Multiple granularities 2). Window size 3). Multiple types of pooling 4). Similarity Measurement: Multiple distance functions

- [**Remarks**]:
    - This model tries to explore multi-perspective method on CNNs, so how to combine the local component I think is the key problem in the experiment, and the author proposed **building block** as a set of groups, which make the whole process clear and efficient.
- [**Experiments TODO**]
    - treat each **Perspective** as **Component** implemented in CNNs.
    - analysis different combination's efficient

    - add the thought of **Semantic Similarity** into the **Model**.

12-NIPS13-Mikolov-Distributed Representations ofWords and Phrases and their Compositionality

PDF, Bib

ICLR13-Mikolov-Efficient estimation of word representations in vector space, Bib

- [**Problem**] the cost of computing the gradient of the Skip-gram model is proportion to the vocabulary size.
- [**Method**]
    - **Hierachical Softmax**
        - Build a binary Huffman tree as the representation of the output layer with W words as its leaves, and for each non-leaves, explicitly represents the relative probabilities of its chiald nodes.
        - reduce to log(W)
    - **Negative Sampling**
        - Assumption: **A good model should be able to differentiate data from noise by means of logistic regression**.
        - replace the objective of nagative sampling, to distingush the target word from draws from the noise distrubution
- [**Others**]
    - **SubSampling of Frequent Words**
        - the most frequent words usually provide less information value than the rare words.
        - discarded with a probability
    - **Phrases Vector**
        - replace words to phrases
        - how to extract phrases? -- words that appear frequently together, and infrequently in other contexts.

This two paper proposed the skip-gram model, and tries to solve two aspects problems: 1). How to make it computable? 2). How to make it more semantical?

As to the first problem, the author tries two method, the one is to replace the softmax as hierachical softmax, which reduce the time complexity to log(W), and the other is to replace the objective with negative sampling.

And when it comes to the second problem, the author tries some tricks. Firstly, he subsampling the frequent words since the vector representations of frequent words do not change significantly. Secondly, he treat phrases as a kind of word to train phrase vectoe, because many phrases have a meaning that is not a simple composition of the meaning of the its individual words.

How to build a model? I think this papers pointed out the right directions.

13-ACL16-Yandex-Siamese CBOW: OptimizingWord Embeddings for Sentence Representations

PDF, Bib, Theono

Siamese CBOW trains word embedding directly for the purpose of being averaged to produce better sentence representation.

Still, a question need to be answered -- How to select negative samples? random? according to similarity? refer to the code! (Update: Both in word2vec[line 442-446] & Siamese CBOW[line 432-448], random slect negative samples!)

- [**Model**]
    - Constructing a supervised training criterion by having our network predict sentences occuring next to each other in the training data, which is similar to **Skip-thought**.
- [**Related Work**]
    - Word2vec
    - Skip-thought
- [**Training Set**]
    - Toronto Book Corpus: 74,004,228 setences; 1,057,070,918 tokens, originating from 7087 unique books.
    - consider tokens appearing 5 times or more, which leads to a vocabulary of 315,643 words.
    - http://www.cs.toronto.edu/~mbweb
- [**TODO**]
    - **replace word2vec in STS with this model**

14-ACL16-Stanford-A Persona-Based Neural Conversation Model

PDF, !Bib

- [**Problem**]: Inconsistent responses via SEQ2SEQ model.

> *message*: How old are you?
> *response*: 16 and you?
> *message*: What's your age?
> *response*: 18.

- [**Model**]
  - Idea: a). Prior Knowledge, b). User Modeling
    - This model represents each individual speaker as a vector or embedding, which encodes speaker-specific information (e.g., dialect, register, age, gender, personal information) that influences the content and style of her response.
    - **Speaker Model**: As in standard SEQ2SEQ models, we first encode message S into a vector representation using the source LSTM. Then for each step in the target side, hidden units are obtained by combining the represeantation produced by the target LSTM at the previous time step, the word representations at the current time step, the speaker embedding.
    - **Speaker-Addressee Model**: replace speaker embedding with an interactive representation by combining two speaker embedding.
- [**Remark**]
  - It is a natural idea to embed our common knowledge or our prior knowledge with a user specific vector representaion and update it throughout the generation process, as human learns from current conversation.
  - **But what do we learn from our daily conversation?**
    - knowledge? thinking is more likely to get it.
    - strength our memory? maybe.
    - interactive? Interaction between two users' knowledge representaion. **How to model?**

15-ACL16-Percy Liang-How Much is 131 Million Dollars Putting Numbers in Perspective with Compositional Descriptions

PDF, !Bib

How to propose a new task? i). Introduction: tell us the Situation, Conflict, Question ii). Problem statement iii). Dataset construction IV). Model V). Evaluation and Results

This work shows how to use crowdworkers to complete a new task. The author is more like a leader of the concert, guide others to complete the fussy task.

- [**Problem**]: How much is 131 million Dollars? About the cost to employ every one in Texas over a lunch period
- [**Model**]: it consists two steps: formula construction and description generation.
  - I). In construction, it composes formulae from numeric facts in a knowledge base and rank the resulting formulas based on familiarity, numeric proximity and semantic compatibility.
  - II). In generation, we convert a formula into natural language using a sequence-to-sequence recurrent neural network.
- [**Data set**]: I). Collecting the knowledge base. II). Collectiong numetic mentions. III). Generating formulas. IV). Collectiong descriptions of formulas. V). Collecting data on formula preference.
- [**Remark**]: How to present number is still a question to answer in NLP. This paper does not directly solve the question, but proposes a new task for how to describe the specific number. Firstly, it constructs some formula. Secondly, it ranks these formulas. Lastly, with sequence-to-sequence model, it transforms the formula to natural language process.

16-ICLR16-facebook-SEQUENCE LEVEL TRAINING WITH RECURRENT NEURAL NETWORKS

PDF, !Bib, Torch

- [**Problem**]: The common approach to solving sequence generation problems is to train the RNN to convert some fixed-size representation of the input into the output sequence, optimizing cross-entropy loss. **There are two main issues with this approach.**
  - **cross-entropy is not what the final model is evaluated on**
  - **during training, the RNN is fed as input the ground-truth**
- [**Problem-Experiment**]: text summary, machine translation, and image captioning
- [**Model**]
  - REINFORCE algorithm for back-propagation on computational graphs that output a probability distribution on actions.
    The base idea is, for each example, if the RNN produces a sequence that yields a reward that is better than average, then push the probability distribution to favor what it decided, and vice-versa - if the reward was worse than average, push the distribution to favor less what it had decided.

  - feed the output of the RNN into itself during training.
    However, the way this is done is by actually taking the k most-likely outputs from the previous step and passing a weighted combination of them as input to the next. The authors motivate this by referring beam search, a method used by others to find a most-likely sequence by expanding out most likely possibilities at each token of the sequence.
- [**Remark**]: I am still not clear yet, I wil read it again. Questions: i). cross-entropy is also not final STS evaluated on, what I should do? ii). Do these two issues happen in STS task? III). What is the problems of the NN models in STS.

TD1-ACL16-Microsoft-Deep Reinforcement Learning with a Natural Language Action Space

PDF, !Bib

Other Papers:

1-ACL16-Simple PPDB: A Paraphrase Database for Simplification

PDF, !Bib

2-harvard-Visual Analysis of Hidden State Dynamics in Recurrent Neural Networks

PDF, code

LSTM Visual Analysis

NAACL

~~A Neural Network-Based Abstract Generation Method to Opinion Summarization~~

~~Bayesian Supervised Domain Adaptation for Short Text Similarity~~

Clustering Paraphrases by Word Sense

Convolutional Neural Networks vs. Convolution Kernels: Feature Engineering for Question Answering

DAG-structured Recurent Neural Networks for Semantic Compositionality

Deep LSTM based Feature Mapping for Query Classification

Dependency Based Embeddings for Sentence Classification Tasks

Dependency Sensitive Convolutional Neural Networks for Modeling Sentences and Documents

~~Hierarchical Attention Networks for Document Classification~~

Learning Distributed Representations of Sentences from Unlabelled Data

Pairwise Word Interaction Modeling with Neural Networks for Semantic Similarity Measurement

Multi-way, Multilingual Neural Machine Translation with a Shared Attention Mechanism