Self-Adaptive Hierarchical Sentence Model

Han Zhao, Zhengdong Lu and Pascal Poupart

UNIVERSITY OF WATERLOO

han.zhao@uwaterloo.ca

July 22, 2015

Outline

Background

- Machine Learning
- Representation Learning

AdaSent

- Motivation
- Architecture
- Learning
- Experiments

Definition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

— Tom M. Mitchell

Definition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

— Tom M. Mitchell

- E – data

Definition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

— Tom M. Mitchell

- E – data
- T – task of interests

Definition

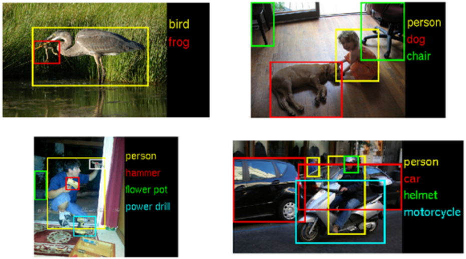
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

— Tom M. Mitchell

- E – data
- T – task of interests
- P – objective function

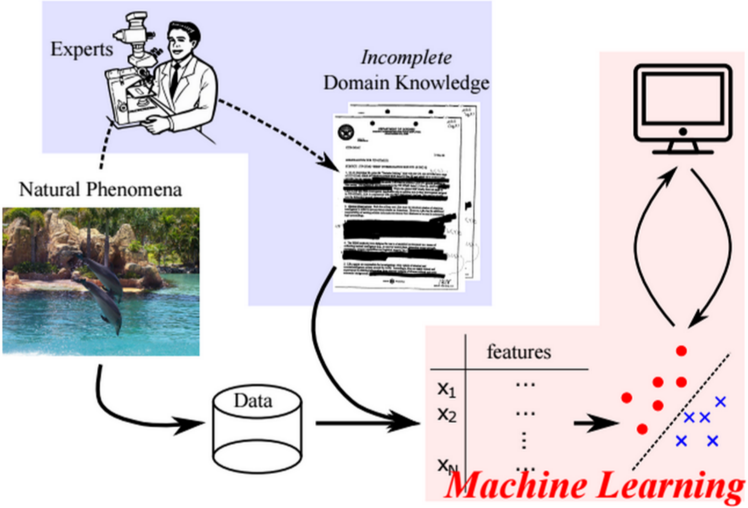
Background

Machine Learning – Application



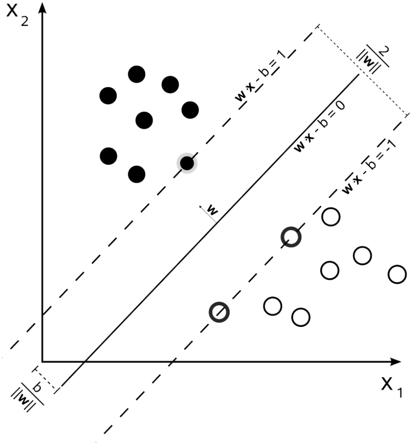
Background

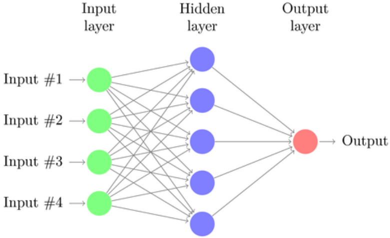
Machine Learning – Pipeline



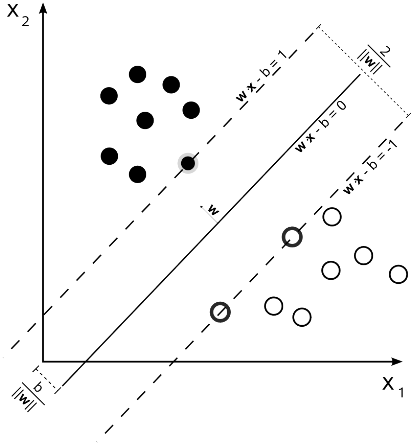
Slide courtesy of Kyunghyun Cho

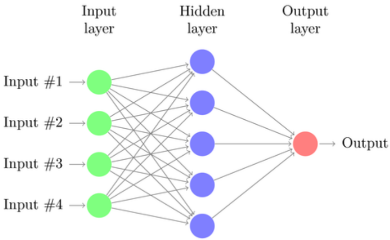
Machine Learning $\approx$ Representation $+$ Objective $+$ Optimization





Machine Learning $\approx$ Representation + Objective + Optimization





Background

Deep Learning

MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2013

Introduction The 10 Technologies Past Years

Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

Background

Deep Learning – Application

- Object Recognition (Krizhevsky et al. 2012)
- Speech Recognition (Graves et al. 2013)
- Neural Machine Translation (Sutskever et al. 2014)
- Face Recognition (Schroff et al. 2015)
- Deep Reinforcement Learning (Mnih et al. 2013)
- Image Caption Generation (Vinyals et al. 2014)
- Text Matching (Hu et al. 2014)
- Text Parsing (Chen et al. 2014)

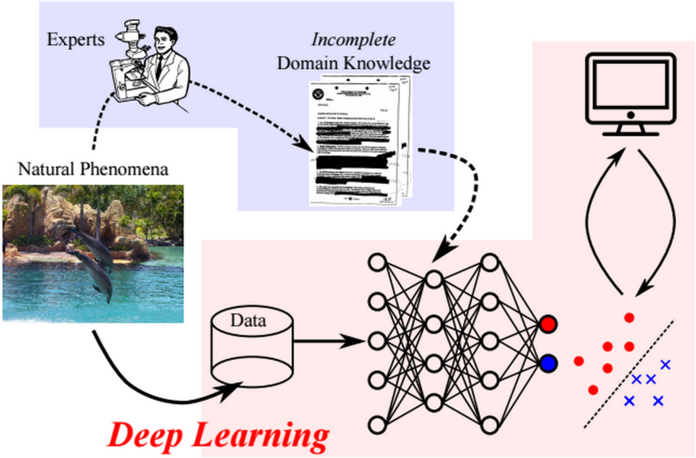
And etc.

Items highlighted in blue happen at Google.

Background

Deep Learning – Representation Learning

Deep Learning: Learning multiple levels of representation directly from massive data

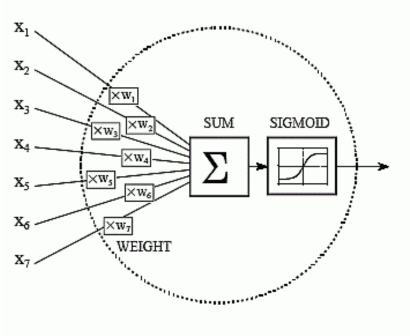


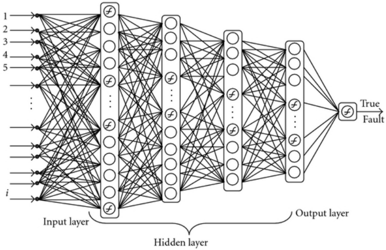
Slide courtesy of Kyunghyun Cho

Background

Deep Learning – Representation Learning

Deep Learning: Learning multiple levels of representation directly from massive data

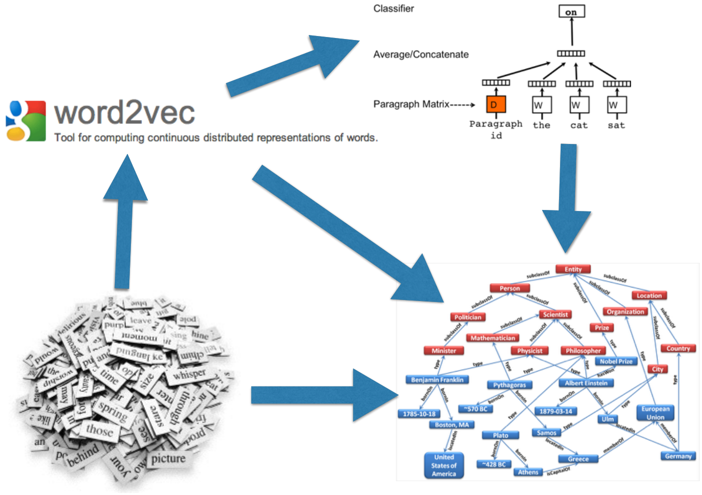




$$a_i^l = \sigma(z_i^l), \quad z_i^l = \sum_j w_{ij} a_j^{l-1}, \quad \sigma(t) = \frac{1}{1 + \exp^{-t}}$$

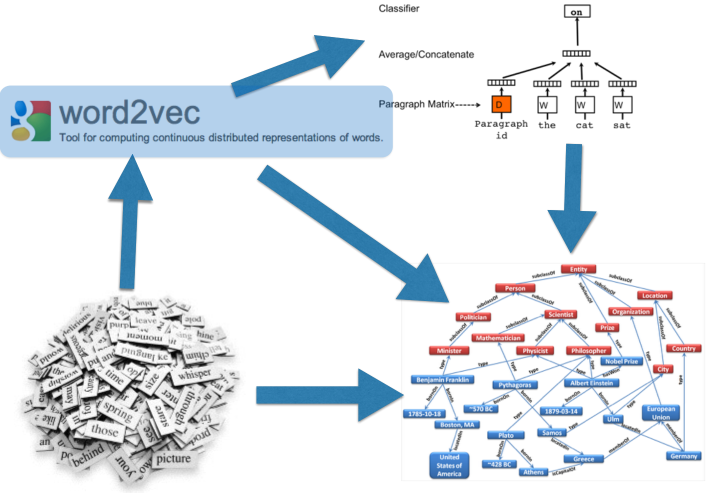
Background

Deep Learning – Natural Language Processing



Background

Deep Learning – Natural Language Processing



Traditional representation of words: One-hot representation.

cat $= [0, 0, 0, 0, 0, 1, 0, 0, \ldots]$

dog $= [0, 1, 0, 0, 0, 0, 0, 0, \ldots]$

Background

Deep Learning – Word Embedding

Traditional representation of words: One-hot representation.
cat $= [0, 0, 0, 0, 0, 1, 0, 0, \ldots]$
dog $= [0, 1, 0, 0, 0, 0, 0, 0, \ldots]$
Pros:

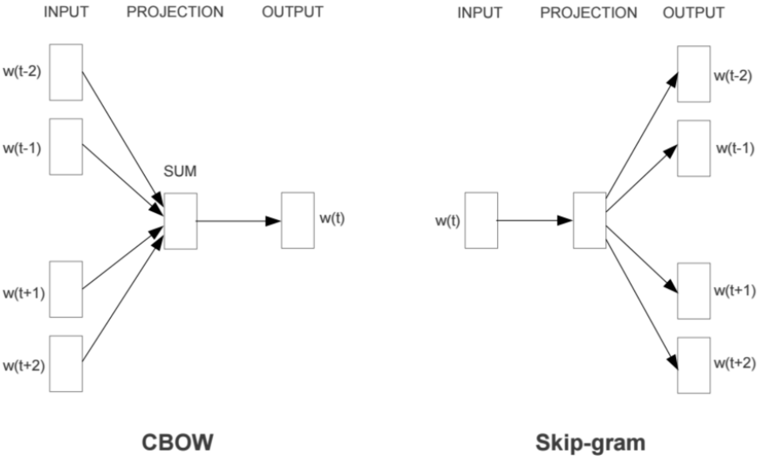
- Simple, intuitive
- Basis of bag-of-words model for document representation

Cons:

- High-dimensional
- No semantic meaning

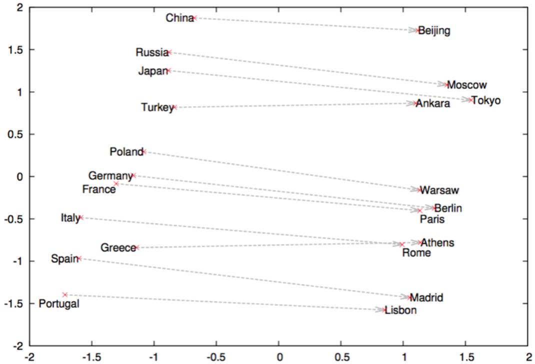
Word2Vec (Minkolov et al)

Distributed word representation: Unsupervised technique to map each word into a dense and real-valued low dimensional vector.



Word2Vec (Minkolov et al)

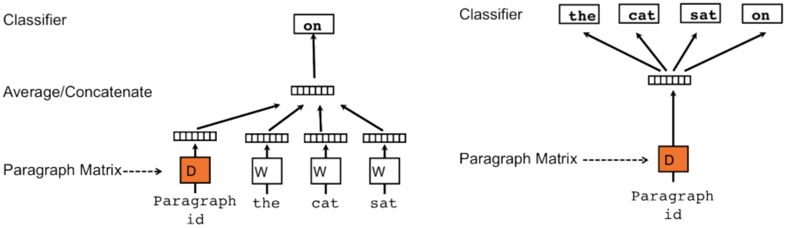
Distributed word representation: Unsupervised technique to map each word into a dense and real-valued low dimensional vector.



$$w(\text{China}) - w(\text{Beijing}) \approx w(\text{Russia}) - w(\text{Moscow}) \approx w(\text{Italy}) - w(\text{Rome})$$

Paragraph Vector (Le et al)

Distributed paragraph representation: Unsupervised technique to map each paragraph into a dense and real-valued low dimensional vector.



Phrase/Sentence/Document Modeling

- Recursive auto-encoder/Matrix-vector recursive neural network (Socher et al. 2011, 2012)
- Convolutional neural network (Kim 2014)
- Dynamic convolutional neural network (Kalchbrenner et al. 2014)
- Recurrent neural network/Bi-directional recurrent neural network (Lai et al. 2015)
- Gated recursive convolutional neural network (Cho et al. 2014)

AdaSent: Self-Adaptive Hierarchical Sentence Model

- Is vector representation with fixed-length enough to represent different granularities of phrases/sentences/documents ?

AdaSent: Self-Adaptive Hierarchical Sentence Model

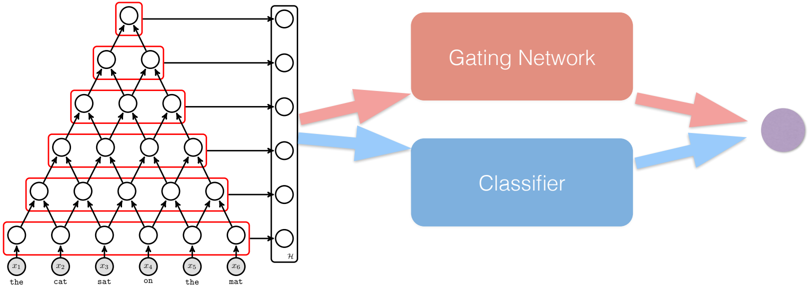
- Is vector representation with fixed-length enough to represent different granularities of phrases/sentences/documents ?
- Can we model the composition behaviour using algebraic operations with enough flexibility ?

AdaSent: Self-Adaptive Hierarchical Sentence Model

- Is vector representation with fixed-length enough to represent different granularities of phrases/sentences/documents ?
- Can we model the composition behaviour using algebraic operations with enough flexibility ?
- Can we design a model which can decide the representation of phrases/sentences on the fly based on the current task at hand ?

AdaSent

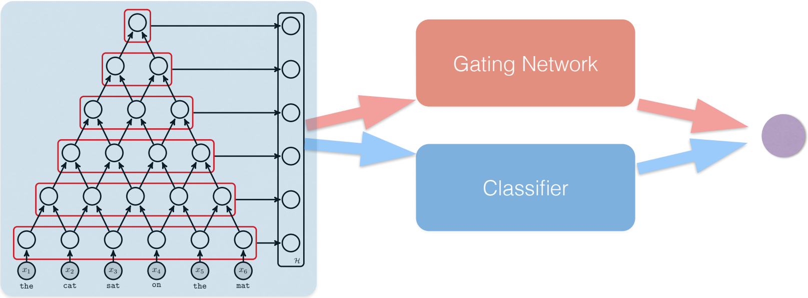
Architecture



Three components:

AdaSent

Architecture

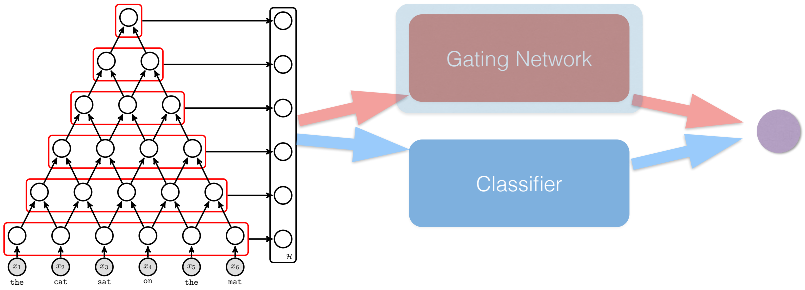


Three components:

- Composition hierarchy

AdaSent

Architecture

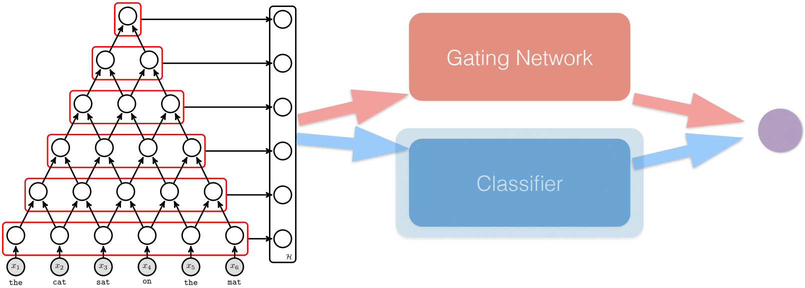


Three components:

- Composition hierarchy
- Gating network

AdaSent

Architecture

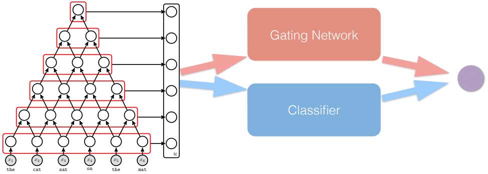


Three components:

- Composition hierarchy
- Gating network
- Classifier

Properties of AdaSent

- Maintains a hierarchy of abstractions from the raw input, rather than a fixed length vector representation
- Implements N-gram model where N ranges from 1 to the length of the sentence
- Implements and extends the mixture-of-experts idea
- Final decision is based on an ensemble of different level of abstractions

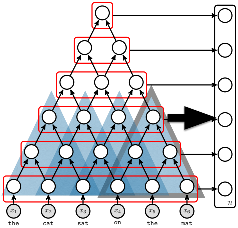


AdaSent

Architecture

Composition Pyramid

Directed acyclic graph whose height depends on the length of input sentence.



Composition dynamics:

$$\begin{cases} h_j^t &= \omega_l h_j^{t-1} + \omega_r h_{j+1}^{t-1} + \omega_c \tilde{h}_j^t \\ \tilde{h}_j^t &= f(W_L h_j^{t-1} + W_R h_{j+1}^{t-1} + b_W) \end{cases}$$

Local combination parametrizations:

$$\begin{pmatrix} \omega_l \\ \omega_r \\ \omega_c \end{pmatrix} = softmax\left(G_L h_j^{t-1} + G_R h_{j+1}^{t-1} + b_G\right)$$

$$softmax\,(\mathbf{v}) = \frac{1}{\sum_{i=1}^{l} \exp(v_i)} \begin{pmatrix} \exp(v_1) \\ \vdots \\ \exp(v_l) \end{pmatrix}$$

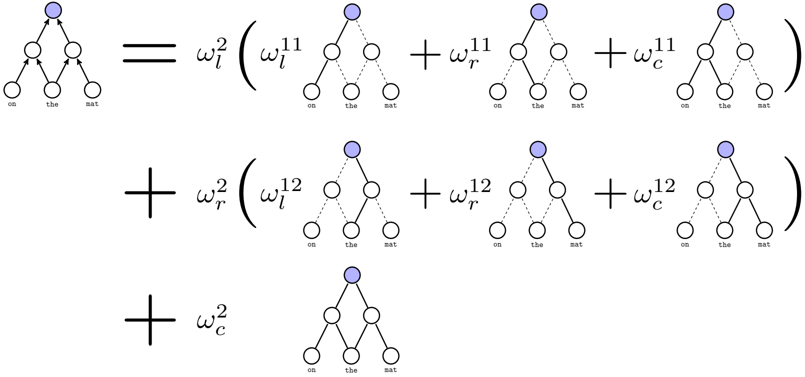
where $W_L, W_R \in \mathbb{R}^{D \times D}$ and $G_L, G_R \in \mathbb{R}^{3 \times D}$.

AdaSent

Architecture

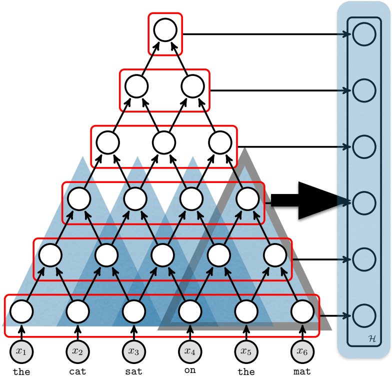
Composition Pyramid

Intuitive interpretation:



Level Pooling

Global (average/max) pooling applied to each level of the pyramid to build the abstraction in the hierarchy.



Average pooling:

$$\bar{h} = \frac{1}{T} \sum_{t=1}^{T} h_t$$

Max pooling:

$$\bar{h}_j = \max_{t \in 1:T} h_{t_j}, \quad \forall j \in 1 : D$$

AdaSent

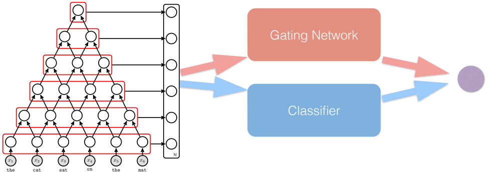
Architecture

Gating Network and Classifier

Gating network: $\omega : \mathbb{R}^D \mapsto \mathbb{R}_+$. Let $\gamma_t \triangleq \omega(\bar{h}_t)$. Constraint: $\sum_{t=1}^{T} \omega(\bar{h}_t) = 1$. Let $g : \mathbb{R}^D \mapsto \Delta_+$ be the classification function.

Classification consensus

$$p(C = c|\mathbf{x}_{1:T}) = \sum_{t=1}^{T} p(c|\mathcal{H}_\mathbf{x} = t) \cdot p(\mathcal{H}_\mathbf{x} = t|\mathbf{x}) = \sum_{t=1}^{T} g_c(\bar{h}_t) \cdot \omega(\bar{h}_t)$$



Backpropagation through Structure (BPTS)

Partial derivative of objective function $\mathcal{L}$ with respect to model parameters:

$$\frac{\partial \mathcal{L}}{\partial W_L} = \sum_{t=1}^{T} \sum_{j=1}^{T-t+1} \frac{\partial \mathcal{L}}{\partial h_j^t} \frac{\partial h_j^t}{\partial W_L}, \frac{\partial \mathcal{L}}{\partial W_R} = \sum_{t=1}^{T} \sum_{j=1}^{T-t+1} \frac{\partial \mathcal{L}}{\partial h_j^t} \frac{\partial h_j^t}{\partial W_R}$$

where

$$\frac{\partial \mathcal{L}}{\partial h_j^t} = \frac{\partial \mathcal{L}}{\partial h_j^{t+1}} \frac{\partial h_j^{t+1}}{\partial h_j^t} + \frac{\partial \mathcal{L}}{\partial h_{j-1}^{t+1}} \frac{\partial h_{j-1}^{t+1}}{\partial h_j^t}$$

$$\frac{\partial h_{j-1}^{t+1}}{\partial h_j^t} = \omega_r I + \omega_c \mathrm{diag}(f') W_R, \frac{\partial h_j^{t+1}}{\partial h_j^t} = \omega_l I + \omega_c \mathrm{diag}(f') W_L$$

Data Sets

- **MR**. Movie reviews data set where each instance is a sentence. The objective is to classify each review by its overall sentiment polarity, either positive or negative.
- **CR**. Annotated customer reviews of 14 products obtained from Amazon. The task is to classify each customer review into positive and negative categories.
- **SUBJ**. Subjectivity data set where the goal is to classify each instance (snippet) as being subjective or objective.
- **MPQA**. Phrase level opinion polarity detection subtask of the MPQA data set.
- **TREC**. Question data set, in which the goal is to classify an instance (question) into 6 different types.

Data Sets

Data	N	**dist**(+,-)	K	$\lvert\mathbf{w}\rvert$	**test**
MR	10662	(0.5, 0.5)	2	18	CV
CR	3788	(0.64, 0.36)	2	17	CV
SUBJ	10000	(0.5, 0.5)	2	21	CV
MPQA	10099	(0.31, 0.69)	2	3	CV
TREC	5952	(0.1,0.2,0.2,0.1,0.2,0.2)	6	10	500

Table: N counts the number of instances and **dist** lists the class distribution in the data set. K represents the number of target classes. $\lvert\mathbf{w}\rvert$ measures the average number of words in each instance. **test** is the size of the test set.

Classification Accuracy

Model	MR	CR	SUBJ	MPQA	TREC
NB-SVM	79.4	81.8	93.2	86.3	-
MNB	79.0	80.0	93.6	86.3	-
RAE	77.7	-	-	86.4	-
MV-RecNN	79.0	-	-	-	-
CNN	81.5	85.0	93.4	89.6	**93.6**
DCNN	-	-	-	-	93.0
P.V.	74.8	78.1	90.5	74.2	91.8
cBoW	77.2	79.9	91.3	86.4	87.3
RNN	77.2	82.3	93.7	90.1	90.2
BRNN	82.3	82.6	94.2	90.3	91.0
GrConv	76.3	81.3	89.5	84.5	88.4
AdaSent	**83.1**	**86.3**	**95.5**	**93.3**	92.4

AdaSent

Experiments

Model Variance

Model	MR	CR	SUBJ
P.V.	71.11 ± 0.80	71.22 ± 1.04	90.22 ± 0.21
cBoW	72.74 ± 1.03	71.86 ± 2.00	90.58 ± 0.52
RNN	74.39 ± 1.70	73.81 ± 3.52	89.97 ± 2.88
BRNN	75.25 ± 1.33	76.72 ± 2.78	90.93 ± 1.00
GrConv	71.64 ± 2.09	71.52 ± 4.18	86.53 ± 1.33
AdaSent	$\mathbf{79.84 \pm 1.26}$	$\mathbf{83.61 \pm 1.60}$	$\mathbf{92.19 \pm 1.19}$
Model	MPQA	TREC	
P.V.	67.93 ± 0.57	86.30 ± 1.10	
cBoW	84.04 ± 1.20	85.16 ± 1.76	
RNN	84.52 ± 1.17	84.24 ± 2.61	
BRNN	85.36 ± 1.13	86.28 ± 0.90	
GrConv	82.00 ± 0.88	82.04 ± 2.23	
AdaSent	$\mathbf{90.42 \pm 0.71}$	$\mathbf{91.10 \pm 1.04}$	

AdaSent

Experiments

Belief Score Distribution

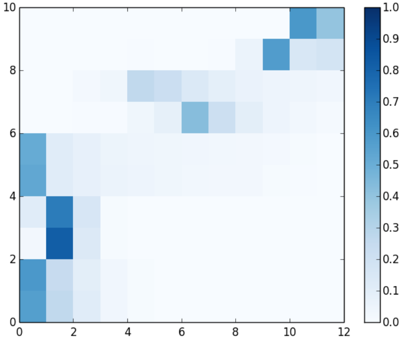
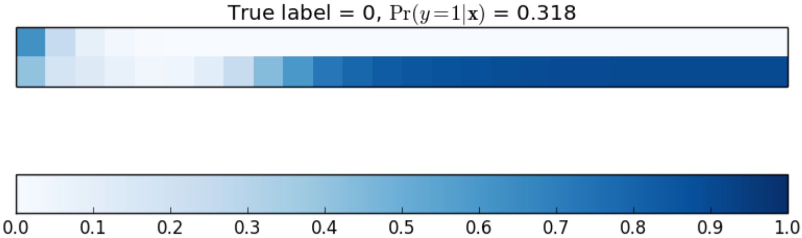


Figure: Each row corresponds to the belief score of a sentence of length 12 sampled from one of the data sets. From top to bottom, the 10 sentences are sampled from MR, CR, SUBJ, MPQA and TREC respectively.

Concrete Example



Sentence: If the movie were all comedy it might work better but it has an ambition to say something about its subjects but not willingness.

Representation Learning - SUBJ

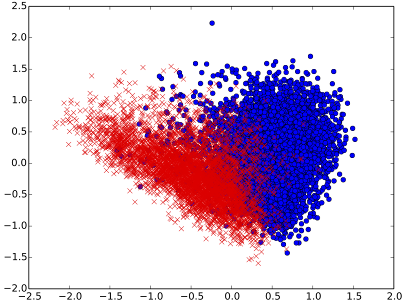


Figure: AdaSent

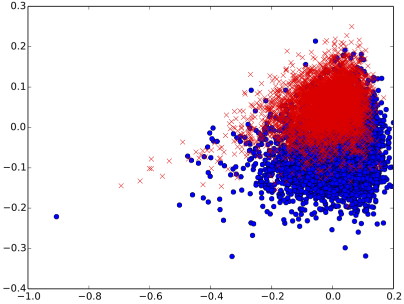


Figure: Original

Representation Learning - MPQA

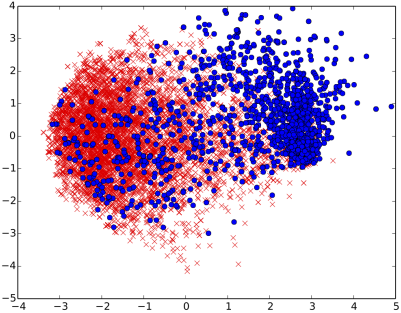


Figure: AdaSent

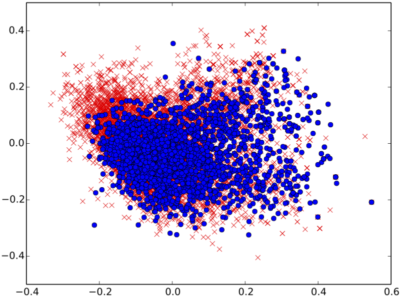


Figure: Original

AdaSent
Experiments

Representation Learning - TREC

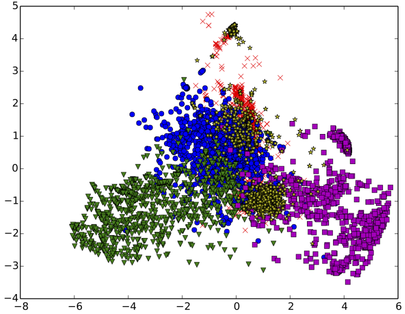


Figure: AdaSent

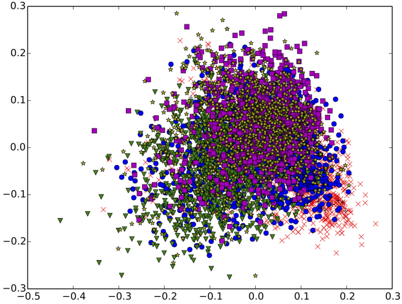


Figure: Original

Thanks

Thanks
Question and Answering
Online Version: arXiv:1504.05070
International Joint Conference on Artificial Intelligence 2015