Text Understanding from Scratch

Xiang Zhang and Yann LeCun

Article presented by Chad DeChant

Journal of Machine Learning Research 12 (2011) 2493-2537 Submitted 1/10; Revised 11/10; Published 8/11

Natural Language Processing (Almost) from Scratch

Ronan Collobert[*] RONAN@COLLOBERT.COM
Jason Weston[†] JWESTON@GOOGLE.COM
Léon Bottou[‡] LEON@BOTTOU.ORG
Michael Karlen MICHAEL.KARLEN@GMAIL.COM
Koray Kavukcuoglu[§] KORAY@CS.NYU.EDU
Pavel Kuksa[¶] PKUKSA@CS.RUTGERS.EDU
NEC Laboratories America
4 Independence Way
Princeton, NJ 08540

Editor: Michael Collins

Abstract

We propose a unified neural network architecture and learning algorithm that can be applied to various natural language processing tasks including part-of-speech tagging, chunking, named entity recognition, and semantic role labeling. This versatility is achieved by trying to avoid task-specific engineering and therefore disregarding a lot of prior knowledge. Instead of exploiting man-made input features carefully optimized for each task, our system learns internal representations on the basis of vast amounts of mostly unlabeled training data. This work is then used as a basis for building a freely available tagging system with good performance and minimal computational requirements.

Keywords: natural language processing, neural networks

Text Understanding from Scratch

Xiang Zhang XIANG@CS.NYU.EDU
Yann LeCun YANN@CS.NYU.EDU
Computer Science Department, Courant Institute of Mathematical Sciences, New York University

Abstract

This article demontrates that we can apply deep learning to text understanding from character-level inputs all the way up to abstract text concepts, using temporal convolutional networks(LeCun et al., 1998) (ConvNets). We apply ConvNets to various large-scale datasets, including ontology classification, sentiment analysis, and text categorization. We show that temporal ConvNets can achieve astonishing performance without the knowledge of words, phrases, sentences and any other syntactic or semantic structures with regards to a human language. Evidence shows that our models can work for both English and Chinese.

1 Introduction

With the advancement of deep learning and availability of large datasets, methods of handling text understanding using deep learning techniques have gradually become available. One technique which draws great interests is word2vec(Mikolov et al., 2013b). Inspired by traditional language models, this technique constructs representation of words into a vector of fixed length trained under a large corpus. Based on the hope that machines may make sense of languages in a formal fashion, many researchers have tried to train a neural network for understanding texts based the features extracted from it or similar techniques, to name a few, (Frome et al., 2013)(Gao et al., 2013)(Le & Mikolov, 2014)(Mikolov et al., 2013a)(Pennington et al., 2014). Most of these techniques try to apply word2vec or similar techniques with an engineered language model.

Paper Highlights

"Text understanding...without artificially embedding knowledge about words, phrases, sentences or any other syntactic or semantic structures associated with a language."

- Input is only characters, not words
- No knowledge of syntax or semantic structures is hardwired in
- Easily modified for other languages

Input

Alphabet size: 69 characters

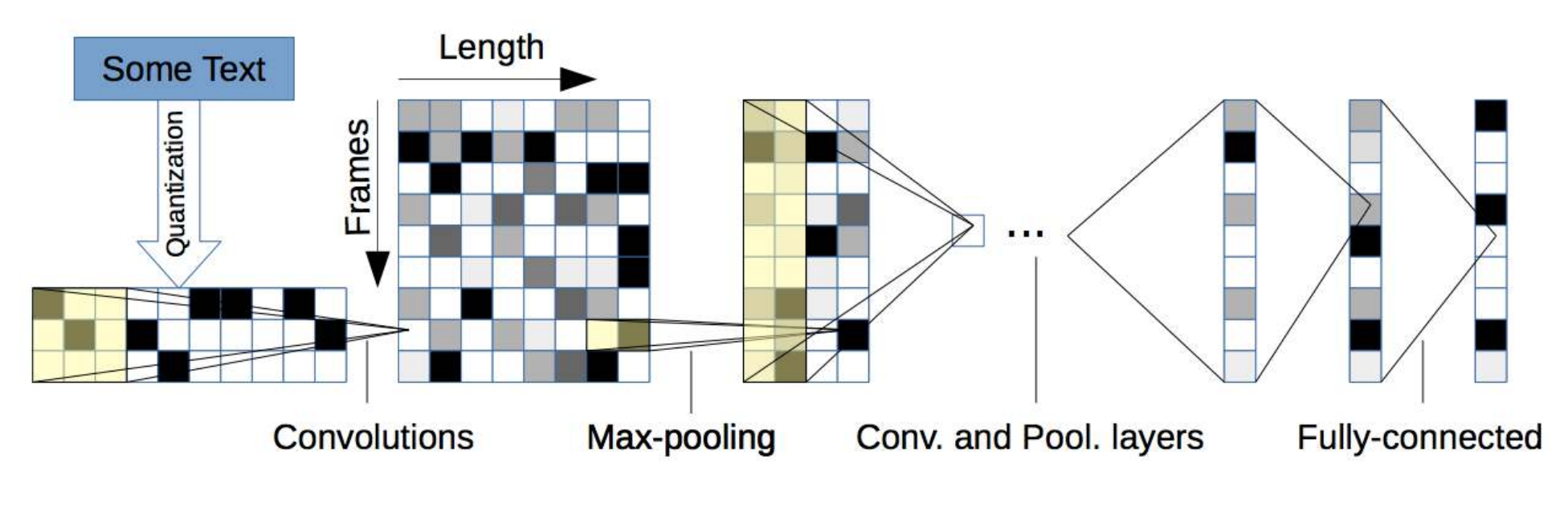
```
abcdefghijklmnopqrst
uvwxyz0123456789-„;.!?:’
”/\|_@#$%^&* ~‘+-=<>()[]{}
```

Length of input = L (1014)

Frame size M is 69

Input is a set of frames of size M x L

ConvNet Design



ConvNet Layers

Convolutional layers

Layer	Large Frame	Small Frame	Kernel	Pool
1	1024	256	7	3
2	1024	256	7	3
3	1024	256	3	N/A
4	1024	256	3	N/A
5	1024	256	3	N/A
6	1024	256	3	3

Fully connected layers

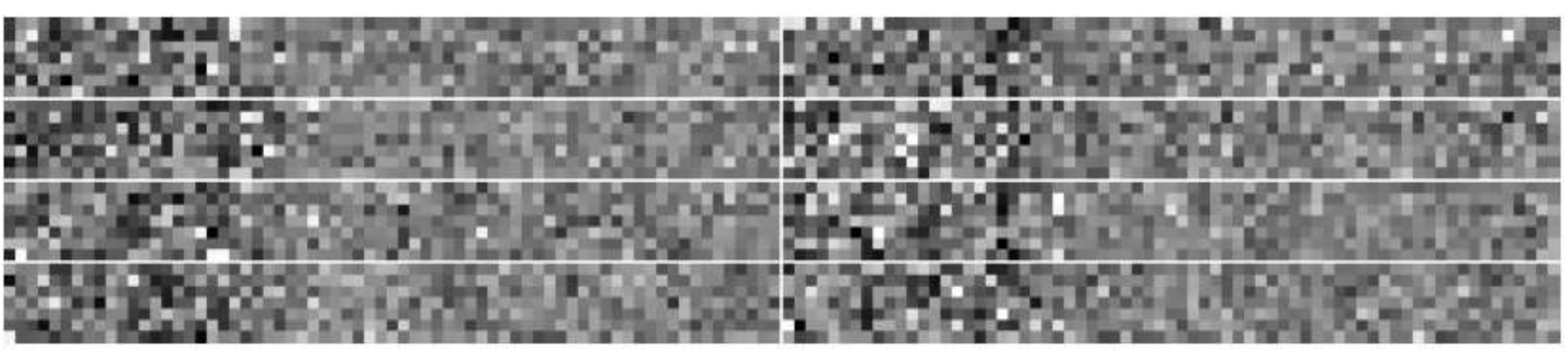
Layer	Output Units Large	Output Units Small
7	2048	1024
8	2048	1024
9	Depends on the problem	

Training

- SGD with minibatch size 128
- Momentum
- Rectified Linear Units
- Torch 7

Learning

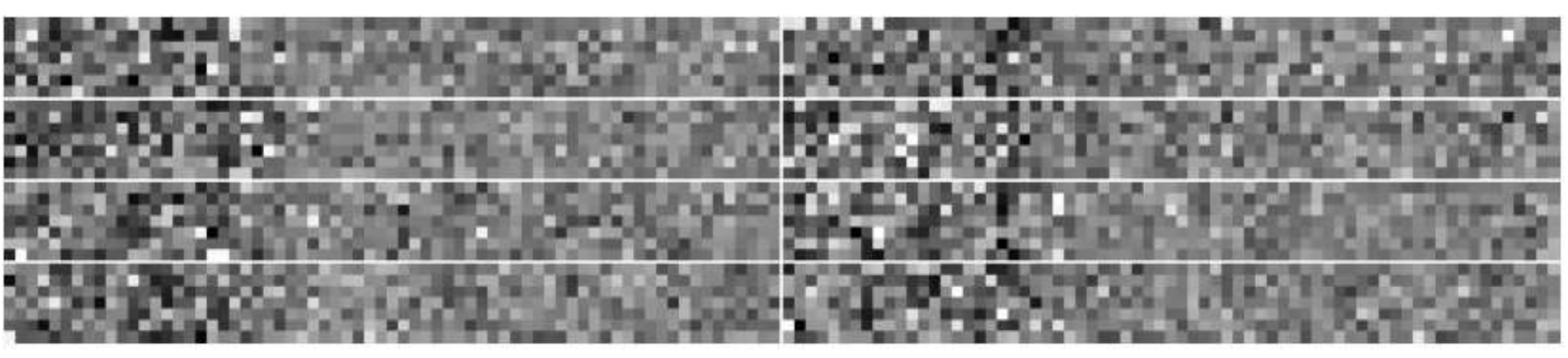
Select kernel weights from the first layer



- Network learned to attach more importance to letters than other characters

Learning

Select kernel weights from the first layer



"We hypothesize that when trained from raw characters, temporal ConvNet is able to learn the hierarchical representations of words, phrases, and sentences in order to understand text."

Data Augmentation with Thesaurus

Improve generalization by increasing the number of training examples

1. Choose r words to be replaced
   $P[r] \sim p^r$
2. Choose the index s in the thesaurus entry of the replacement word
   $P[s] \sim q^s$

$q = p = 0.5$

geometric distribution

Dataset and Results

"The unfortunate fact in [the] literature is that there is no openly accessible dataset that is large enough or with labels of sufficient quality for us..."

Dataset and Results

Several new datasets for:

- Sentiment analysis
- text categorization
- ontology classification

Comparisons

Performance comparisons only against their own implementations of:

- Bag of Words

  Most common 5000 words from each dataset

- word2vec

  Same 5000 vectors trained on Google news corpus used for all dataset comparisons

→ Less than state of the art comparisons

Amazon review sentiment analysis

	Total	Chosen	Full	Polarity
1	2,746,559	2,206,886	1,250,000	2,200,000
2	1,791,219	1,290,278	1,250,000	1,250,000
3	2,892,566	1,975,014	1,250,000	0
4	6,551,166	4,576,293	1,250,000	1,250,000
5	20,705,260	16,307,871	1,250,000	2,200,000

→ A very large dataset

Input text: Amazon reviews between 100 and 1000 characters

Amazon review results

Table 6. Result on Amazon review full score dataset. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	**93.73%**	**73.28%**
Large ConvNet	Yes	83.67%	71.37%
Small ConvNet	No	82.10%	70.12%
Small ConvNet	Yes	84.42%	68.18%
Bag of Words	No	52.13%	51.93%
word2vec	No	38.22%	38.25%

Table 7. Result on Amazon review polarity dataset. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	**99.71%**	**96.34%**
Large ConvNet	Yes	99.51%	96.08%
Small ConvNet	No	98.24%	95.84%
Small ConvNet	Yes	98.57%	96.01%
Bag of Words	No	88.46%	85.54%
word2vec	No	75.15%	73.07%

Amazon review results

Table 6. Result on Amazon review full score dataset. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	**93.73%**	73.28%
Large ConvNet	Yes	83.67%	71.37%
Small ConvNet	No	82.10%	70.12%
Small ConvNet	Yes	84.42%	68.18%
Bag of Words	No	52.13%	51.93%
word2vec	No	38.22%	38.25%

Table 7. Result on Amazon review polarity dataset. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	**99.71%**	96.34%
Large ConvNet	Yes	99.51%	96.08%
Small ConvNet	No	98.24%	95.84%
Small ConvNet	Yes	98.57%	96.01%
Bag of Words	No	88.46%	85.54%
word2vec	No	75.15%	73.07%

Other results for comparison: movie sentiment analysis

Classifier	Fine-grained (%)	Binary (%)
NB	41.0	81.8
BiNB	41.9	83.1
SVM	40.7	79.4
RecNTN	45.7	85.4
Max-TDNN	37.4	77.1
NBoW	42.4	80.5
DCNN	**48.5**	**86.8**

From Kalchbrenner, Grefenstette, Blunsome, "A Convolutional Neural Network for Modeling Sentences" 2014

Yahoo answers topic dataset

Table 8. Yahoo! Answers topic classification dataset

Category	Total	Train	Test
Society & Culture	295,340	140,000	5,000
Science & Mathematics	169,586	140,000	5,000
Health	278,942	140,000	5,000
Education & Reference	206,440	140,000	5,000
Computers & Internet	281,696	140,000	5,000
Sports	146,396	140,000	5,000
Business & Finance	265,182	140,000	5,000
Entertainment & Music	440,548	140,000	5,000
Family & Relationships	517,849	140,000	5,000
Politics & Government	152,564	140,000	5,000

Input text: Question title, question text, best answer

Yahoo Answers results

Table 9. Results on Yahoo! Answers dataset. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	71.76%	69.84%
Large ConvNet	Yes	**72.23%**	**69.92%**
Small ConvNet	No	70.10%	69.92%
Small ConvNet	Yes	70.73%	69.81%
Bag of Words	No	66.75%	66.44%
word2vec	No	58.84%	59.01%

Yahoo Answers results

Table 9. Results on Yahoo! Answers dataset. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	71.76%	69.84%
Large ConvNet	Yes	**72.23%**	**69.92%**
Small ConvNet	No	70.10%	69.92%
Small ConvNet	Yes	70.73%	69.81%
Bag of Words	No	66.75%	66.44%
word2vec	No	58.84%	59.01%

Other results for comparison: 6-way question classification

Classifier	Features	Acc. (%)
HIER	unigram, POS, head chunks NE, semantic relations	91.0
MAXENT	unigram, bigram, trigram POS, chunks, NE, supertags CCG parser, WordNet	92.6
MAXENT	unigram, bigram, trigram POS, wh-word, head word word shape, parser hypernyms, WordNet	93.6
SVM	unigram, POS, wh-word head word, parser hypernyms, WordNet 60 hand-coded rules	95.0
MAX-TDNN	unsupervised vectors	84.4
NBOW	unsupervised vectors	88.2
DCNN	unsupervised vectors	93.0

From Kalchbrenner, Grefenstette, Blunsome, "A Convolutional Neural Network for Modelling Sentences" 2014

DBpedia Ontology Classification

Class	Total	Train	Test
Company	63,058	40,000	5,000
Educational Institution	50,450	40,000	5,000
Artist	95,505	40,000	5,000
Athlete	268,104	40,000	5,000
Office Holder	47,417	40,000	5,000
Mean Of Transportation	47,473	40,000	5,000
Building	67,788	40,000	5,000
Natural Place	60,091	40,000	5,000
Village	159,977	40,000	5,000
Animal	187,587	40,000	5,000
Plant	50,585	40,000	5,000
Album	117,683	40,000	5,000
Film	86,486	40,000	5,000
Written Work	55,174	40,000	5,000

Input text: title and abstract. length ≤ 1014 characters

DBpedia Ontology Results

Table 4. DBpedia results. The numbers are accuracy.

Model	Thesaurus	Train	Test
Large ConvNet	No	**99.95%**	98.26%
Large ConvNet	Yes	99.81%	**98.40%**
Small ConvNet	No	99.70%	97.99%
Small ConvNet	Yes	99.64%	98.15%
Bag of Words	No	96.62%	96.43%
word2vec	No	89.64%	89.41%

News categorization results

Category	Total	Train	Test
World	81,456	40,000	1,100
Sports	62,163	40,000	1,100
Business	56,656	40,000	1,100
Sci/Tech	41,194	40,000	1,100

Model	Thesaurus	Train	Test
Large ConvNet	No	99.00%	91.12%
Large ConvNet	Yes	**99.00%**	**91.64%**
Small ConvNet	No	98.94%	89.32%
Small ConvNet	Yes	98.97%	90.39%
Bag of Words	No	88.35%	88.29%
word2vec	No	85.30%	85.28%

Input text: title of article and description, length ≤ 1014 chars

News categorization in Chinese

Extend the model to work with Chinese:

- Segment text:

  我常常跟朋友看电影

  ioftenseemovieswithfriends

  → 我 常常 跟 朋友 看 电影

  i often see movies with friends

- transliterate:

  → wo3 chang2chang2 gen1 peng2you3 kan4 dian4ying3

News categorization in Chinese

Table 12. Sogou News dataset

Category	Total	Train	Test
Sports	645,931	150,000	10,000
Finance	315,551	150,000	10,000
Entertainment	160,409	150,000	10,000
Automobile	167,647	150,000	10,000
Technology	188,111	150,000	10,000

Table 13. Result on Sogou News corpus. The numbers are accuracy

Model	Thesaurus	Train	Test
Large ConvNet	No	**97.64%**	**97.05%**
Small ConvNet	No	97.45%	97.03%
Bag of Words	No	95.69%	95.46%

Input text: title of article and content, $100 \leq \text{length} \leq 1014$ chars

Conclusions & Speculations

- Good results
- End to end learning
- New datasets

Conclusions & Speculations

Journal of Machine Learning Research 12 (2011) 2493-2537 Submitted 1/10; Revised 11/10; Published 8/11

Natural Language Processing (Almost) from Scratch

Ronan Collobert[*] RONAN@COLLOBERT.COM
Jason Weston[†] JWESTON@GOOGLE.COM
Léon Bottou[‡] LEON@BOTTOU.ORG
Michael Karlen MICHAEL.KARLEN@GMAIL.COM
Koray Kavukcuoglu[§] KORAY@CS.NYU.EDU
Pavel Kuksa[¶] PKUKSA@CS.RUTGERS.EDU
NEC Laboratories America
4 Independence Way
Princeton, NJ 08540

Editor: Michael Collins

Abstract

We propose a unified neural network architecture and learning algorithm that can be applied to various natural language processing tasks including part-of-speech tagging, chunking, named entity recognition, and semantic role labeling. This versatility is achieved by trying to avoid task-specific engineering and therefore disregarding a lot of prior knowledge. Instead of exploiting man-made input features carefully optimized for each task, our system learns internal representations on the basis of vast amounts of mostly unlabeled training data. This work is then used as a basis for building a freely available tagging system with good performance and minimal computational requirements.

Keywords: natural language processing, neural networks

Conclusions & Speculations

Reinventing the wheel?

> "Text understanding...without artificially embedding knowledge about words, phrases, sentences or any other syntactic or semantic structures associated with a language."

Thank you