NLP Applications using Deep Learning

Giri Iyengar

Cornell University

gi43@cornell.edu

Feb 26, 2018

Agenda for the day

- Sentiment Analysis
- Neural Machine Translation
- Entailment

Overview

1. Sentiment Analysis
2. Neural Machine Translation
3. Textual Entailment

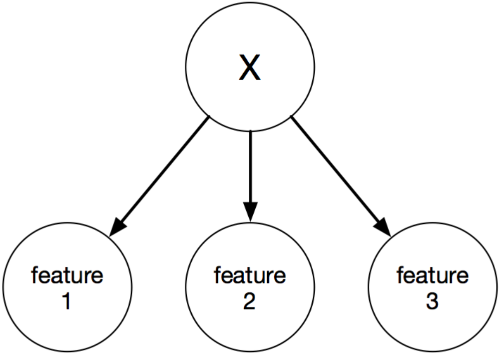
Sentiment Analysis

Defining Sentiment Analysis

Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral. It is also known as opinion mining, deriving the opinion or attitude of a speaker.

- Common Use case: How people feel about a particular {Brand, Topic, News, Company, Product}?

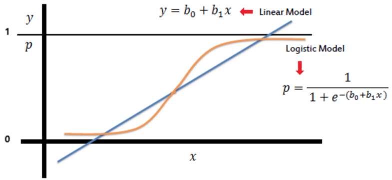
Sentiment Analysis: Naive Bayes

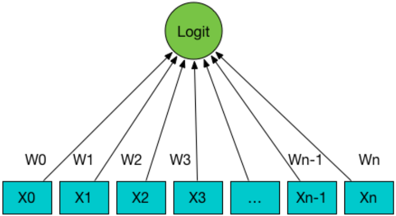


Bag of words

Use words without caring for their **order** as features. For each word, use its **vector** representation (e.g. word2vec, GloVe)

Sentiment Analysis: Logistic Regression





Sentiment Analysis using Bag of Words

- Context is very important
- Word order conveys information
- Redundancy in human language

Recurrent Neural Networks for Sentiment Analysis

- Word order is naturally captured in this model

Recurrent Neural Networks for Sentiment Analysis

- Word order is naturally captured in this model
- Entire sentence/phrase is considered

Recurrent Neural Networks for Sentiment Analysis

- Word order is naturally captured in this model
- Entire sentence/phrase is considered
- Variable lengths of inputs are naturally handled without artificial padding / chopping

Sentiment Analysis: RNN

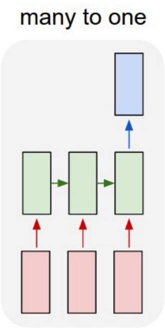


Figure: Source: A. Karpathy

Many to One RNN

Feed words of a sentence in and then run a classifier on the final representation

Sentiment Analysis: RNN

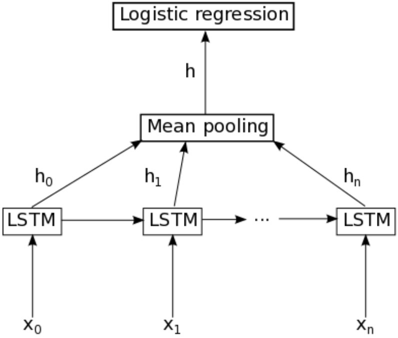


Figure: Source: deeplearning.net

Average pooling over time

Feed words of a sentence in and then average pooling over time to classify

Sentiment Analysis: CNN

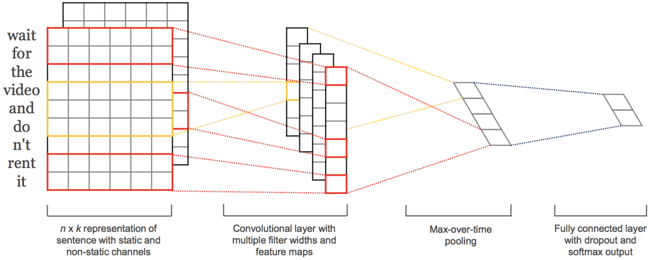


Figure: Source: Y. Kim, NYU

Sentiment Analysis using CNN

Start with sentence matrix. Consider **k** words at a time, where **k** varies. Pool and classify.

Sentiment Analysis using CNNs and RNNs

- Both CNN and RNN can use static word representations

Sentiment Analysis using CNNs and RNNs

- Both CNN and RNN can use static word representations
- Or, they could *learn/fine-tune* these representations

Sentiment Analysis using CNNs and RNNs

- Both CNN and RNN can use static word representations
- Or, they could *learn/fine-tune* these representations
- Can make use of multiple representations. Not just one

Sentiment Analysis using CNNs and RNNs

- Both CNN and RNN can use static word representations
- Or, they could *learn/fine-tune* these representations
- Can make use of multiple representations. Not just one
- Winner has been switching places time and again

Sentiment Analysis using Recursive Networks

- Both CNNs and RNNs seem to not capture the *compositional* structure in language

Sentiment Analysis using Recursive Networks

- Both CNNs and RNNs seem to not capture the *compositional* structure in language
- Historically, sentences have been represented by *Computational Linguists* using Parse Trees

Sentiment Analysis using Recursive Networks

- Both CNNs and RNNs seem to not capture the *compositional* structure in language
- Historically, sentences have been represented by *Computational Linguists* using Parse Trees
- These trees seem to be a more natural representation of a sentence structure

Sentiment Analysis using Recursive Networks

- Both CNNs and RNNs seem to not capture the *compositional* structure in language
- Historically, sentences have been represented by *Computational Linguists* using Parse Trees
- These trees seem to be a more natural representation of a sentence structure
- Can we build models that explore/exploit this structure?

Parse Trees

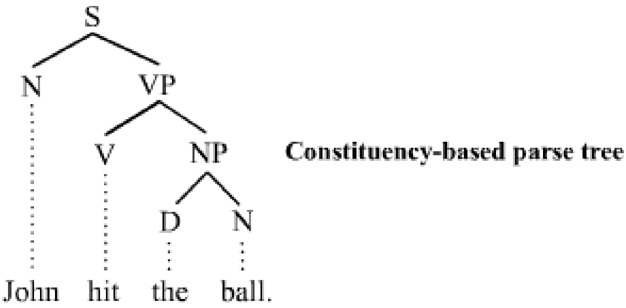
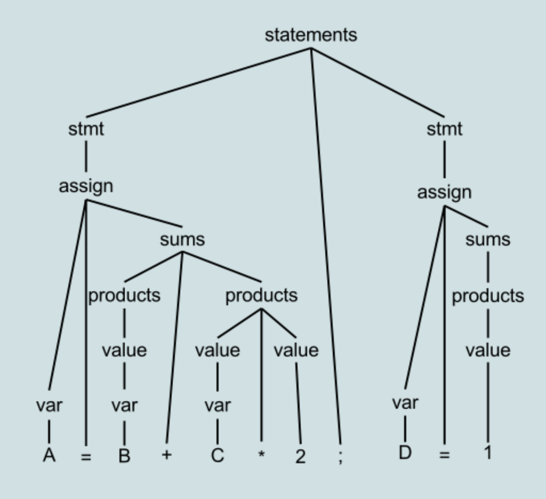


Figure: Parse Tree examples: Courtesy Wikipedia

Parse Trees


statements
stmt
stmt
assign
assign
sums
sums
products
products
products
value
value
value
value
var
var
var
var
A
=
B
+
C
*
2
;
D
=
1

Recursive Networks for Sentiment Analysis

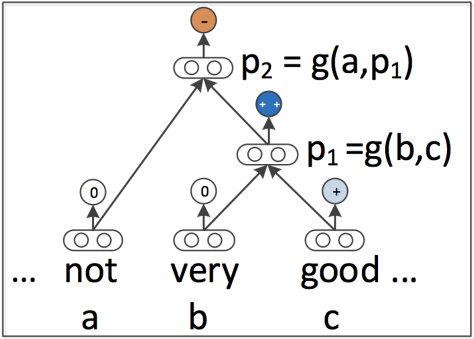


Figure: Source: Richard Socher/EMNLP2013

Overview

1. Sentiment Analysis
2. Neural Machine Translation
3. Textual Entailment

Neural Machine Translation using RNN

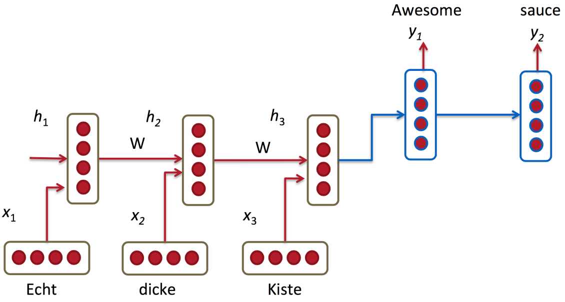


Figure: Image Source: http://cs224d.stanford.edu/lectures/CS224d-Lecture8.pdf

Neural Machine Translation using RNN

- Feed entire sentence in **source** language

Neural Machine Translation using RNN

- Feed entire sentence in **source** language
- Extract sentence in **target** language

Neural Machine Translation using RNN

- Feed entire sentence in **source** language
- Extract sentence in **target** language
- Train using *parallel corpora*

Neural Machine Translation using RNN

- Feed entire sentence in **source** language
- Extract sentence in **target** language
- Train using *parallel corpora*
- E.g. European Parliamentary proceedings. Canadian Hansard

Neural Machine Translation using RNN

- Feed entire sentence in **source** language
- Extract sentence in **target** language
- Train using *parallel corpora*
- E.g. European Parliamentary proceedings. Canadian Hansard
- Encoder-Decoder paradigm

Beam Search to generate target sentences

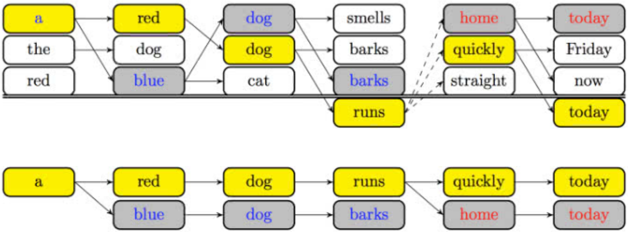


Figure: Image Source:
https://deepage.net/img/beam-search/beam-search-opt.jpg

Neural Machine Translation using RNN

- The entire sentence is *squished* into one vector
- Increasing the hidden state size is not a solution

Attention based NMT models

- While translating, have access to the entire hidden state
- Each word translation looks up the words emitted so far *and parts* of the source sentence
- Model identifies parts of the source sentence that the decoder pays *attention* to while translating

Attention based NMT model

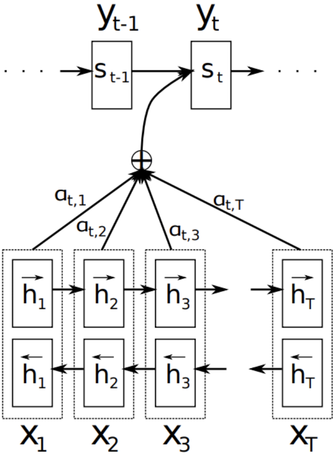


Figure: Image Source: Bahdanau, ICLR 2015

Overview

1. Sentiment Analysis
2. Neural Machine Translation
3. Textual Entailment

Textual Entailment

- **Textual Entailment** is directional relation between text fragments
- Entailing and Entailed sentences are referred as *text t* and *hypothesis h* respectively
- Measures *Natural Language Understanding* because it requires a semantic interpretation of the text

Textual Entailment

- An example of a **positive TE** (text entails hypothesis) is:
  - *text*: If you help the needy, God will reward you.
  - *hypothesis*: Giving money to a poor man has good consequences.
- An example of a **negative TE** (text contradicts hypothesis) is:
  - *text*: If you help the needy, God will reward you.
  - *hypothesis*: Giving money to a poor man has no consequences.
- An example of a **non-TE** (text does not entail nor contradict) is:
  - *text*: If you help the needy, God will reward you.
  - *hypothesis*: Giving money to a poor man will make you a better person.


thank you
danke
謝謝
ngiyabonga
teşekkür ederim
gracias
merci
obrigado
спасибо
Баярлалаа
рахмат
bedankt
dziękuję
hvala
mauruuru
dank je
tapadh leat
mochchakkeram
go raibh maith agat
sukriya
kop khun krap
terima kasih
grazie
arigatō
takk
dakujem
мерси
감사합니다
ευχαριστώ
ありがとう
sagolun
taiku
shukriya
dhanyavadagalu
tanemirt
rahmet
diolch
xièxie
najis tuke
kam sah hamnida
mesi
enkosi
sobodi
dėkuji
kiitos
dankie
dhanyavad
faafetai lava
vinaka
blagodaram
mersi
barka
welalin
tack
spas
misaotra
matondo
paldies
grazzi
mahalo
хвала
asante
manana
obrigada
murakoze
tenki
mamnun
дякую
djiere dieuf
chnorakaloutioun
gratias ago
gràcies
sulpay
dankon
aciu
akun
köszönöm
gracie
bayarlalaa
nandi
nanni
trugarez
merce