Word Embeddings

ADVISOR: PROF. AMITABHA MUKERJEE

PRANJAL SINGH

10327511

Index

1. Motivation
2. Past
3. Objectives
4. Earlier Work
    a. NLP from Scratch
    b. Bilingual Word Embeddings
    c. Vector Space Word Representation
5. Dataset
6. Results (English and Hindi)
7. Conclusion and Future Work

Motivation

- Word embeddings have the power to capture syntax and semantics both
- We have many sources of unsupervised raw data but not supervised data
- Unsupervised techniques could greatly improve existing supervised systems **(Collobert et al.(2013))**

Leveraging large amount of data floating around, we can improve existing systems

Past

- LSA and LDA were used to capture word embeddings(not exactly) and hence derive semantic relations

- Most of the existing systems treat word as atomic units

BUT

Words also inherit meanings which can only be defined if we represent it as a vector/combination of latent words

Objective

➡ To maximize probability of raw text given a context window

So for a given context window of size *c*:

$$\max \frac{1}{T} \sum_{t=1}^{T} \log p \left(w_t | w_{t-c}^{t+c} \right)$$

Earlier Work

- word2vec (Mikolov et al., 2013) learns embeddings using neural language model
- Collobert & Weston, 2011 : NLP from Scratch
- Bilingual Word Representations (Zou et al. al & Manning et al., 2013)

Embeddings

Word2vec

1) **CBOW**

Embeddings are represented by a set of latent variables and initialized randomly

Training learns these for each word w_t in the vocabulary

So for a given context window of size *c*:

$$\max \frac{1}{T} \sum_{t=1}^{T} \log p\left(w_t | w_{t-c}^{t+c}\right)$$

$$p(w_t | w_{t-c}^{t+c}) = \frac{\exp\left({e'_{w_t}}^\top \cdot \sum_{-c \le j \le c, j \ne 0} e_{w_{t+j}}\right)}{\sum_w \exp\left({e'_w}^\top \cdot \sum_{-c \le j \le c, j \ne 0} e_{w_{t+j}}\right)}$$

Embeddings

Word2vec

2) **Relational Constraint Model**

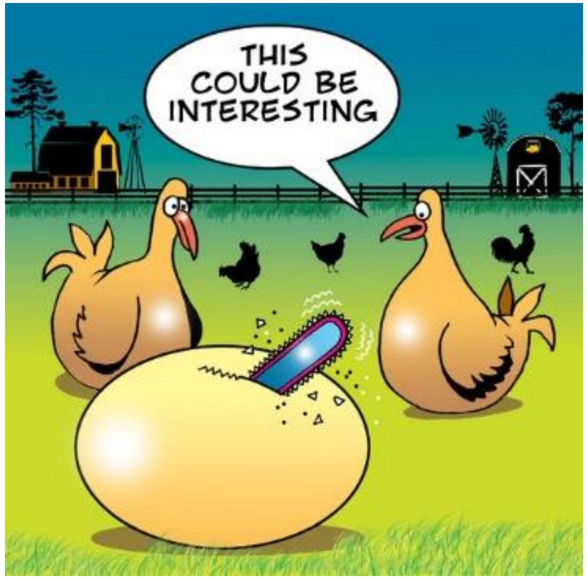
Define R as a set of relation between two words and relations have scores associated to indicate strength

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{w\in\mathbf{R}_{w_i}}\log p\left(w|w_i\right),$$

--- They do not include scores of these relations

Interesting!!!!

A joint model:



$$\max \frac{1}{T} \sum_{t=1}^{T} \log p\left(w_t | w_{t-c}^{t+c}\right)$$

\+

$$\frac{1}{N} \sum_{i=1}^{N} \sum_{w \in \mathbf{R}_{w_i}} \log p\left(w | w_i\right),$$

NLP from Scratch

- Built a unified architecture for tasks such as POS tagging, Chunking, NER
- Compared against classical NLP benchmarks
- Avoided task specific engineering
- Generalize a system to handle multiple tasks

NLP from Scratch

- Learn lookup table by back propagation
- Words are mapped to *d-dimensional* vector using lookup table operation
- Lookup table returns a matrix for a given sentence

NLP from Scratch

Used entire English Wikipedia to learn word embeddings (631 million words)

Tokenized using Penn Treebank Tokenizer

The total training time was about four weeks

Window size: 11 and a Hidden layer with 100 units

Objective: Seek a network that computes a higher score when given a legal phrase than when given an incorrect phrase

$$\theta \mapsto \sum_{x \in \mathcal{X}} \sum_{w \in \mathcal{D}} \max \left\{ 0 \,,\, 1 - f_\theta(x) + f_\theta(x^{(w)}) \right\}$$

NLP from Scratch

FRANCE 454	JESUS 1973	XBOX 6909	REDDISH 11724	SCRATCHED 29869	MEGABITS 87025
AUSTRIA	GOD	AMIGA	GREENISH	NAILED	OCTETS
BELGIUM	SATI	PLAYSTATION	BLUISH	SMASHED	MB/S
GERMANY	CHRIST	MSX	PINKISH	PUNCHED	BIT/S
ITALY	SATAN	IPOD	PURPLISH	POPPED	BAUD
GREECE	KALI	SEGA	BROWNISH	CRIMPED	CARATS
SWEDEN	INDRA	PSNUMBER	GREYISH	SCRAPED	KBIT/S
NORWAY	VISHNU	HD	GRAYISH	SCREWED	MEGAHERTZ
EUROPE	ANANDA	DREAMCAST	WHITISH	SECTIONED	MEGAPIXELS
HUNGARY	PARVATI	GEFORCE	SILVERY	SLASHED	GBIT/S
SWITZERLAND	GRACE	CAPCOM	YELLOWISH	RIPPED	AMPERES

Bilingual Word Embeddings

- It proposes a method to learn bilingual embeddings rather than just monolingual embeddings

- So it utilizes counts of MT alignments derived from Berkeley aligner to initialize monolingual embeddings of another language

$$W_{t\text{-}init} = \sum_{s=1}^{S} \frac{C_{ts} + 1}{C_t + S} W_s$$

- They have used the same formulation as Collobert et al.(2008) to learn embeddings except that they have used global context information as in Huang et al.(2012)

Bilingual Word Embeddings

- Their objective function captures information of both monolingual embedding and also on translation matrices, also called alignment matrices

- They have trained on 100K-vocabulary word embeddings

- With 500,000 iterations it took 19 days of training on 8-core machine

- For phrase similarity in 2 languages, they have averaged out the word embedding vectors corresponding to each word in both phrases and then taken cosine similarity to quantize amount of semantic similarity

Dataset

- Hindi :Wikipedia text dump (279MB)
- English: Wikipedia text dump (95MB)

Result (English)

"boy" is to "father" as "girl" is to ...?

(Top 3)

1.	Mother	0.6219688653945923
2.	Grandmother	0.5560075640678406
3.	Wife	0.5442352890968323

Result (English)

he his	**she:?**
big bigger	**bad:?**
going went	**being:?**

- 'he' is to 'his' as 'she' is to **‘her’**
- 'big' is to 'bigger' as 'bad' is to **‘worse’**
- 'going' is to 'went' as 'being' is to **‘were’**

Result (English)

Which word doesn't go with the others?

breakfast **cereal** **dinner** **lunch**

- **cereal**

Result (Hindi)

भारत

यूक्रेन	0.488481163979
मैक्सिको	0.472263723612
फिलीपीन्स	0.461070656776
कोसोवो	0.445656210184
कैलिफोर्निया	0.438328802586
तिरुवनंतपुरम	0.437484622002
ओंटारियो	0.437374174595
सिचुआन	0.436686635017
लम्पुर	0.436174809933
वेलेस्ले	0.434365183115

Result (Hindi)

Odd one out

'भारत'

'रूस'

'मुम्बई'

'चीन'

'मुम्बई'

Result (Hindi)

x= similar(['भारत'.decode('utf8')], topn=5)

प्रदेश	0.434905201197
देश	0.434299349785
तिब्बत	0.434264868498
आन्ध्रप्रदेश	0.428886473179
लद्दाख़	0.427965015173

Result (Hindi)

x= similar(['व्यापार'.decode('utf8')], topn=5)

व्यवसाय	0.671647787094
पुनर्बीमा	0.617935776711
वाणिज्य	0.612713575363
संस्थागत	0.61127692461
बैंकिंग	0.607060432434

Result (Hindi)

कम

0.013972 0.020021 0.005228 0.001282 -0.096880 -0.064957 -0.004378 0.057942 -0.109471 -0.052513 -0.002228 0.068519 0.117182 0.009550 0.008309 -0.035241 0.042594 0.046013 0.022055 0.033392 -0.046861 0.083555 0.003501 0.032369 -0.051409 0.042281 0.060196 0.016986 0.023544 0.014908 -0.095546 0.010151 -0.028563 -0.079369 0.045530 -0.002945 -0.023547 -0.058014 -0.038463 0.083010 -0.028450 0.018251 0.005231 -0.006079 -0.005987 -0.000233 0.066247 0.021251 -0.041221 -0.002379 0.064932 -0.080568 -0.113520 -0.053706 0.042745 0.021324 -0.086906 0.030630 -0.068239 -0.119651 0.027618 -0.029169 0.048726 -0.017188

Future Work

What if we **ADD** the embeddings???

Or

If we **SUBTRACT** the embeddings??

Very Big Bigger

Such phrases and words should have greater semantic similarity

Can operations such as addition/subtraction give a better insight into such relationships (applicable for Hindi also)

Future Work

Indian Cricketer Sachin

Infact above phrase and word may belong to same embedding

Future Work

- The embeddings obtained could help in initializing the embeddings used in work of Collobert and Weston
- Manning et al.(2013) have used semantic information to improve word embeddings
- Collobert et al.(2008) have used large unlabeled data to do the same thing.
- **Can we use syntactic or morphological information to improve word embeddings or even produce some good word embeddings ?**

Motivation

- Morphologically similar words have some sought of close connection between them
- e.g. morphology, phonology, etymology

References

- Manning, W. Y. (2013). Bilingual Word Embeddings for Phrase-Based Machine Translation. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, (p. 2013).

- Weston, R. C. (2011). Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research, 12*, 2493--2537.

- Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3781.

- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani & K. Weinberger (ed.), *Advances in Neural Information Processing Systems 26* (pp. 3111--3119) .

- Zheng, X., Chen, H. & Xu, T. (2013). Deep Learning for Chinese Word Segmentation and POS Tagging.. *EMNLP* (p./pp. 647-657), : ACL. ISBN: 978-1-937284-97-8

- Sarath Chandar A P, Mitesh M Khapra, Ravindran B, Vikas Raykar, Amrita Saha, "Multilingual Deep Learning". In Deep Learning Workshop at NIPS 2013.

Thank You!!!