Unsupervised Most Frequent Sense Determination Using Word Embeddings

Sudha Bhingardive
Research Scholar,
IIT Bombay, India.

Supervisor
Prof. Pushpak Bhattacharyya

Roadmap

- Introduction: Most Frequent Sense Baseline
- Approach
  - Word Embeddings
  - Creating Sense Embeddings
  - Detecting MFS
- Experiments and Results
- MFS for Indian Languages
- Conclusion and Future Work

Most Frequent Sense: WSD Baseline

- Assigns the most frequent sense to every content words in the corpus
- Context is not considered while assigning senses
- For example: cricket [S1 : game sense S2: insect sense]
- If MFS (cricket) = S1
  - A boy is playing **cricket_S1** on the playground
  - **Cricket_S1** bites won't hurt you
  - **Cricket_S1** singing in the home is a sign of good luck

Motivation

- An acid test for any new Word Sense Disambiguation (WSD) algorithm is its performance against the Most Frequent Sense (MFS)
  - For many unsupervised WSD algorithm this MFS baseline is also a skyline

- Getting MFS values requires sense annotated corpus in enormous amounts

Our Approach [UMFS–WE]

- An unsupervised approach for MFS detection using word embeddings
  - Does not require any hand-tagged text

- Word *embedding* of a word is compared with *sense embeddings* to obtain the MFS sense with the highest similarity.

- Domain independent approach and can be easily ported across multiple languages

Word Embeddings

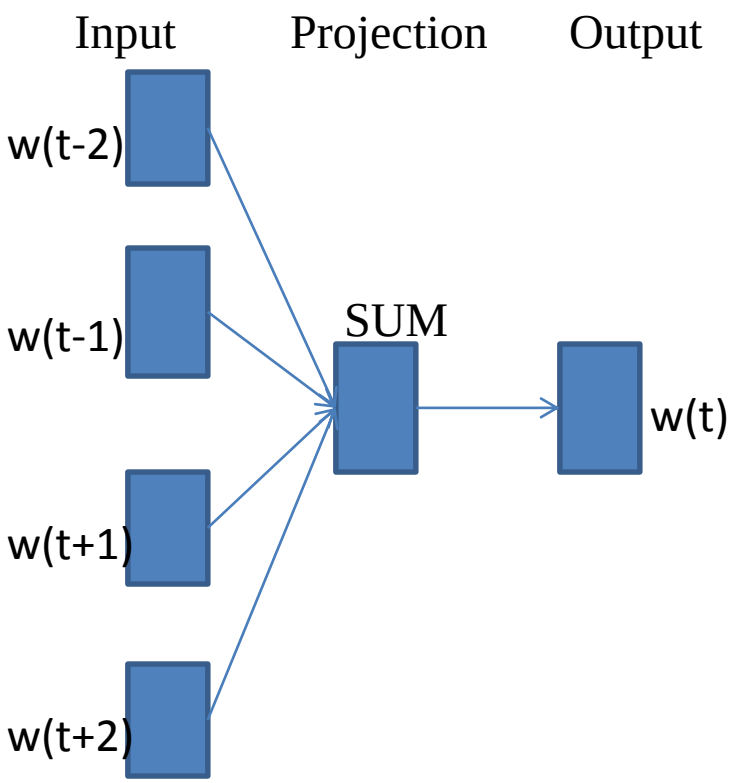
- Represent each word with low-dimensional real valued vector.

- Increasingly being used in variety of Natural Language Processing tasks.

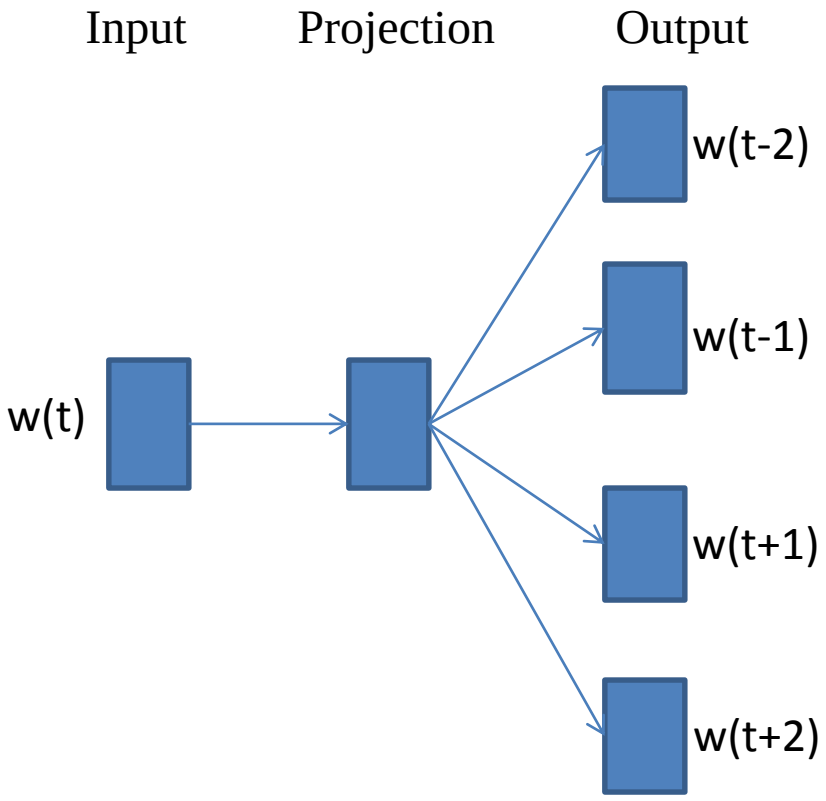
Word Embeddings Tool

- **word2vec tool (Mikolov et. al, 2013)**
  - One of the most popular word embedding tool
  - Source code provided
  - Pre-trained embeddings provided
  - Based on distributional hypothesis

Word Embeddings Tool contd..



Continuous bag of words model (CBOW)



Skip-gram model

Word Embeddings Tool contd..

- **word2vec tool** (Mikolov et. al, 2013)

  - It captures many linguistic regularities
  - Vector(king') –Vector('man')+Vector['woman']=> Vector('queen')

Sense Embeddings

- Sense embeddings are obtained by taking the average of word embeddings of each word in the sense-bag

$$vec(S_i) = \frac{\sum_{x \in SB(S_i)} vec(x)}{N}$$

- S_i - ith sense of a word W
- N - Number of words present in the sense-bag $SB(S_i)$
- The sense-bag for the sense S_i is created as below,

$$\text{SB}(S_i) = \{\text{x} | \text{x - Features}(S_i)\}$$

- $\text{Features}(S_i)$ - WordNet based features for sense S_i

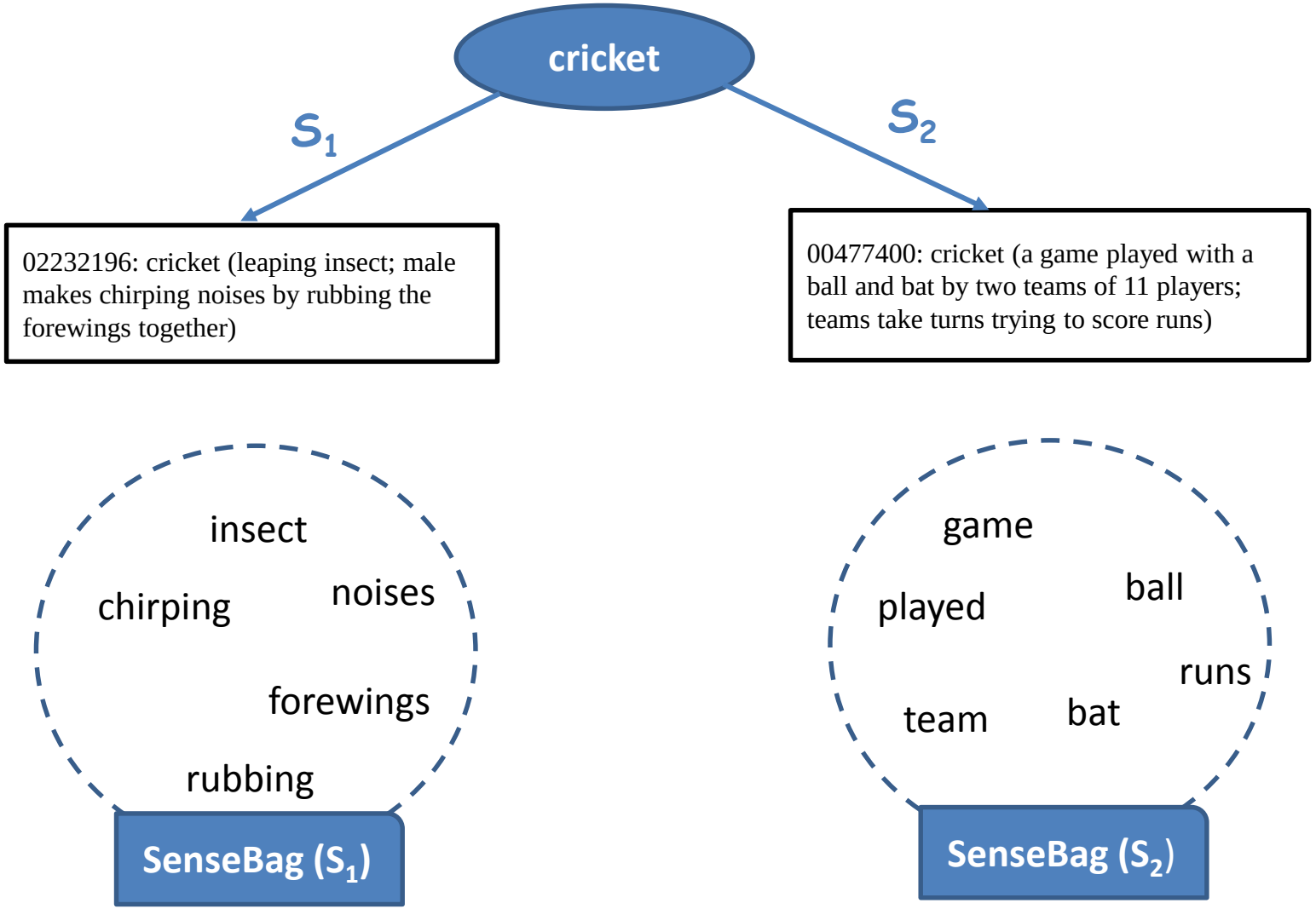
MFS Detection

- We treat the MFS identification problem as finding the closest cluster centroid (*i.e.*, sense embedding) with respect to a given word.
- Cosine similarity is used.
- Most frequent sense is obtained by using the following formulation,

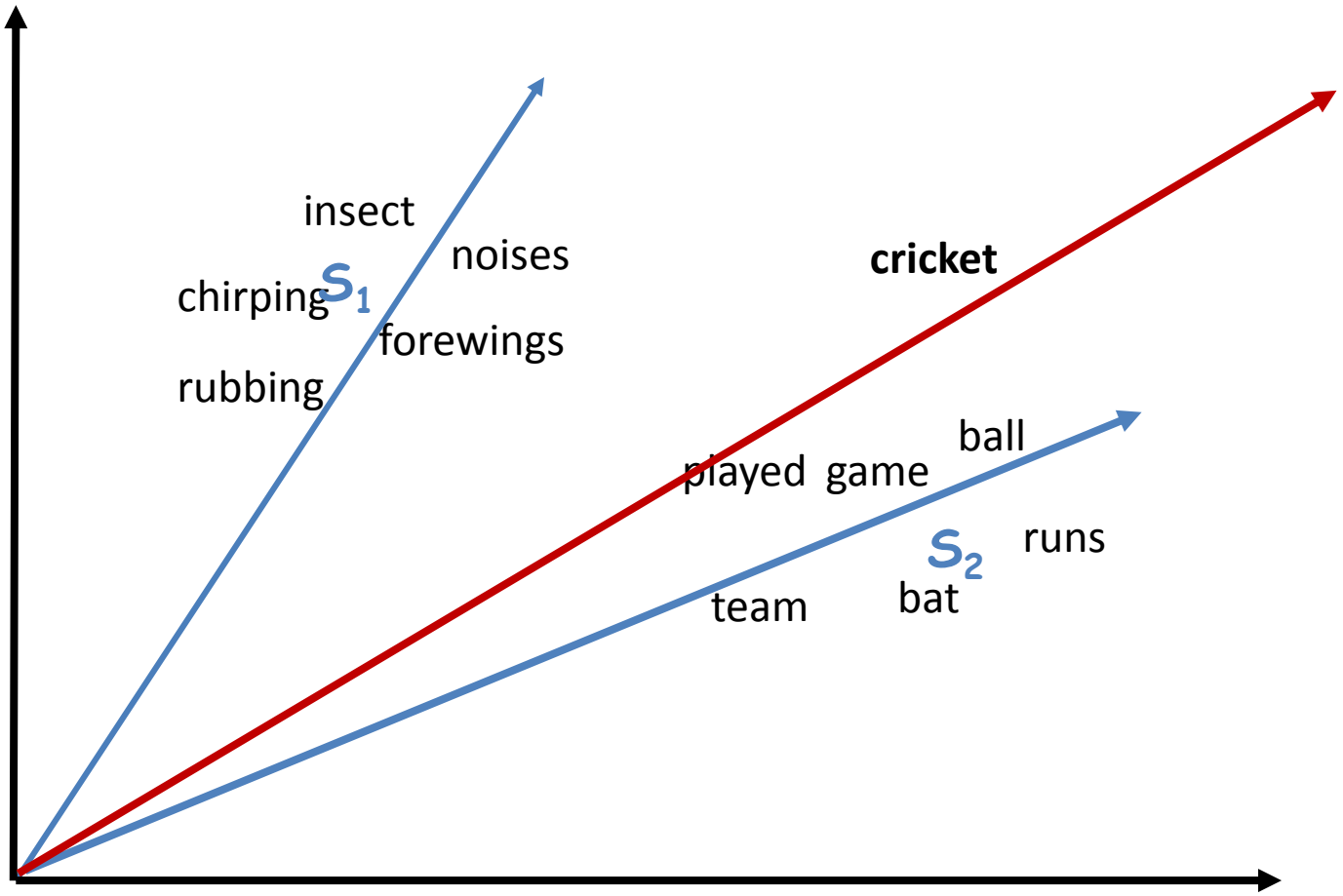
$$MFS_w = \underset{S_i}{\text{argmax}} \cos(vec$$

  - $vec(W)$ - word embedding of a word W
  - S_i- ith sense of word W
  - $vec(S_i)$ - sense embedding for S_i

MFS Detection



MFS Detection contd..



Experiments

A. Experiments on WSD
   1. Experiments on WSD using Skip-Gram model
      - Hindi (Newspaper)
      - English (SENSEVAL-2 and SENSEVAL-3)
   2. Experiments on WSD using different word vector models
   3. Comparing WSD results using different sense vector models
      - Retrofitting Sense Vector Model (English)
   4. Experiments on WSD for words which do not exists in SemCor

B. Experiments on selected words (34 polysemous words from SENSEVAL-2 corpus)
   1. Experiments using different word vector models
   2. Comparing results with various sizes of vector dimensions

Experiments

A. Experiments on WSD

1. Experiments on WSD using Skip-Gram model
   - Hindi (Newspaper)
   - English (SENSEVAL-2 and SENSEVAL-3)

[A.1] Experiments on WSD using skip-gram model

- Training of word embeddings:
  - Hindi: Bojar (2014) corpus (44 M sentences)
  - English: Pre-trained Google-News word embeddings
- Datasets used for WSD:
  - Hindi: Newspaper dataset
  - English: SENSEVAL-2 and SENSEVAL-3
- Experiments are restricted to polysemous nouns.

[A.1] Results on Hindi WSD

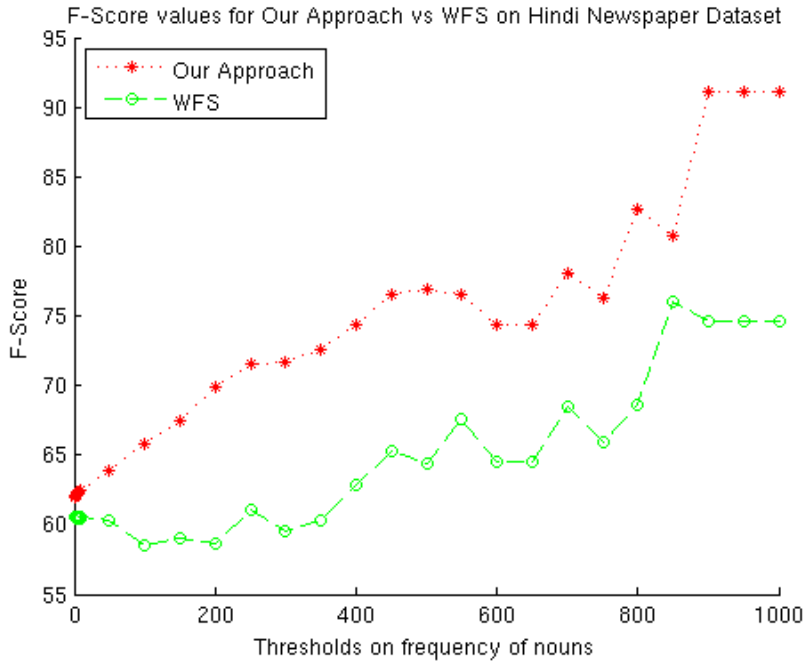
HINDI WSD	Newspaper dataset		
	Precision	Recall	F-Score
UMFS-WE	62.43	61.58	62.00
WFS	61.73	59.31	60.49

[A.1] Results on English WSD

ENGLISH WSD	SENSEVAL-2 dataset			SENSEVAL-3 dataset		
	Precision	Recall	F-Score	Precision	Recall	F-Score
UMFS-WE	52.39	52.27	52.34	43.34	43.22	43.28
WFS	61.72	58.16	59.88	66.57	64.89	65.72

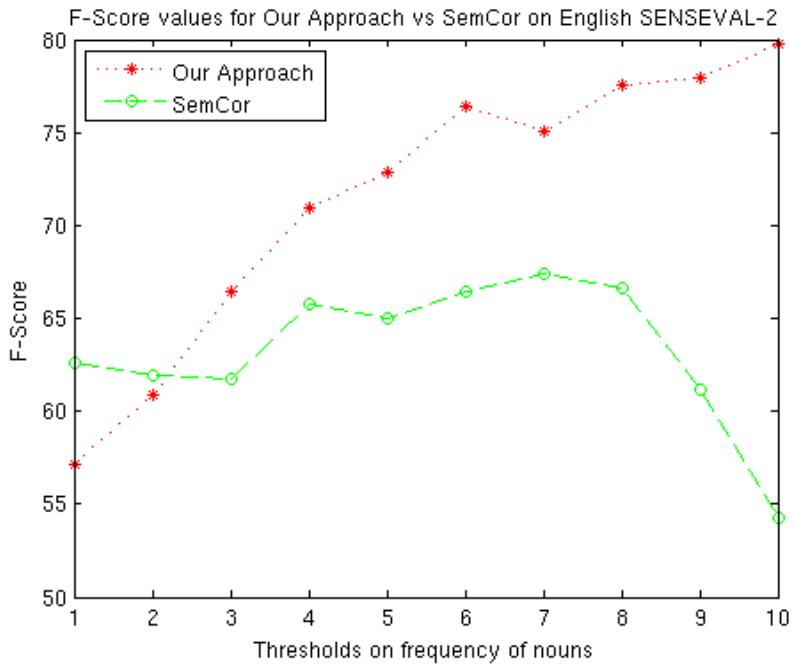
[A.1] Results on WSD contd..

- F-Score is also calculated for increasing thresholds on the frequency of nouns appearing in the corpus.



Hindi WSD

[A.1] Results on WSD contd..



English WSD

[A.1] Results on WSD contd..

- WordNet feature selection for sense embeddings creation

Sense Vectors Using WordNet features	Precision	Recall	F-measure
SB	51.73	38.13	43.89
SB+GB	53.31	52.39	52.85
SB+GB+EB	56.61	55.84	56.22
SB+GB+EB+PSB	59.53	58.72	59.12
SB+GB+EB+PGB	60.57	59.75	60.16
SB+GB+EB+PEB	60.12	59.3	59.71
SB+GB+EB+PSB+PGB	57.59	56.81	57.19
SB+GB+EB+PSB+PEB	58.93	58.13	58.52
SB+GB+EB+PGB+PEB	**62.43**	**61.58**	**62**
SB+GB+EB+PSB+PGB+PEB	58.56	57.76	58.16

SB: Synset Bag
GB: Gloss Bag
EB: Example Bag
PSB: Parent Synset Bag
PGB: Parent Gloss Bag
PEB: Parent Example Bag

Table: Hindi WSD results using various WordNet features for Sense Embedding creation

Experiments

A. Experiments on WSD

1. Experiments on WSD using Skip-Gram model
   - Hindi (Newspaper)
   - English (SENSEVAL-2 and SENSEVAL-3)
2. Experiments on WSD using different word vector models

[A.2] Experiments on WSD using various Word Vector models

- We compared MFS results on various word vector models which are listed below:

Word Vector Model	Dimensions
SkipGram-Google-News (Mikolov et. al, 2013)	300
Senna (Collobert et. al, 2011)	50
MetaOptimize (Turian et. al, 2010)	50
RNN (Mikolov et. al, 2011)	640
Glove (Pennington et. al, 2014)	300
Global Context (Huang et. al, 2013)	50
Multilingual (Faruqui et.al, 2014)	512
SkipGram-BNC (Mikolov et. al, 2013)	300
SkipGram-Brown (Mikolov et. al, 2013)	300

[A.2] Experiments on WSD using various Word Vector models contd..

WordVector	Noun	Adj	Adv	Verb
SkipGram-Google-News	54.49	**50.56**	**47.66**	20.66
Senna	54.49	40.44	28.97	21.9
RNN	39.07	28.65	40.18	19.42
MetaOptimize	33.73	36.51	32.71	19.83
Glove	**54.69**	49.43	39.25	18.18
Global Context	48.3	32.02	31.77	20.66
SkipGram-BNC	53.03	48.87	39.25	**23.14**
SkipGram-Brown	30.29	48.87	27.10	13.29

Table: English WSD results for words with corpus frequency > 2

Experiments

A. Experiments on WSD

1. Experiments on WSD using Skip-Gram model
   - Hindi (Newspaper)
   - English (SENSEVAL-2 and SENSEVAL-3)
2. Experiments on WSD using different word vector models
3. Comparing WSD results using different sense vector models
   - Retrofitting Sense Vector Model (Jauhar et al, 2015)

[A.3] Results on WSD

WordVector	SenseVector	Noun	Adj	Adv	Verb
SkipGram-Google-News	Our model	**58.87**	53.53	**46.34**	20.49
	Retrofitting	47.84	**57.57**	32.92	**21.73**
Senna	Our model	**61.29**	43.43	**21.95**	**24.22**
	Retrofitting	6.9	**68.68**	**21.95**	1.86
RNN	Our model	**42.2**	26.26	**40.24**	**21.11**
	Retrofitting	10.48	**62.62**	21.95	1.24
MetaOptimize	Our model	**37.9**	50.5	**31.7**	**18.01**
	Retrofitting	10.48	**62.62**	21.95	1.24
Glove	Our model	**58.33**	53.33	**39.02**	**17.39**
	Retrofitting	9.94	**62.62**	21.95	1.24
Global Context	Our model	53.22	37.37	**24.39**	19.25
	Retrofitting	12.36	**68.68**	21.95	1.24
SkipGram-Brown	Our model	29.31	60.6	**23.17**	11.42
	Retrofitting	11.49	**68.68**	21.95	1.26

Table: English WSD results for words with corpus frequency > 2

Experiments

A. Experiments on WSD

1. Experiments on WSD using Skip-Gram model
   - Hindi (Newspaper)
   - English (SENSEVAL-2 and SENSEVAL-3)
2. Experiments on WSD using different word vector models
3. Comparing WSD results using different sense vector models
   - Retrofitting Sense Vector Model (English)
4. Experiments on WSD for words which do not exists in SemCor

[A.4] English WSD results for SEMEVAL-2 words which do not exist in SemCor

Word Vector	F-score
SkipGram-Google-News	**84.12**
Senna	79.67
RNN	24.59
MetaOptimize	22.76
Glove	79.03
Global Context	28.09
Multilingual	35.48
SkipGram-BNC	68.29
SkipGram-BNC+Brown	74.79

proliferate, agreeable, bell_ringer, audacious, disco, delete, prestigious, option, peal, impaired, ringer, flatulent, unwashed, cervix, discordant, eloquently, carillon, full-blown, incompetence, stick_on, illiteracy, implicate, galvanize, retard, libel, obsession, altar, polyp, unintelligible, governance, bell_ringing.

Experiments

A. Experiments on WSD
   1. Experiments on WSD using Skip-Gram model
      - Hindi (Newspaper)
      - English (SENSEVAL-2 and SENSEVAL-3)
   2. Experiments on WSD using different word vector models
   3. Comparing WSD results using different sense vector models
      - Retrofitting Sense Vector Model (English)
   4. Experiments on WSD for words which do not exists in SemCor

B. Experiments on selected words (34 polysemous words from SENSEVAL-2 corpus)
   1. Experiments using different word vector models

[B.1] Experiments on selected words

- 34 polysemous nouns, where each one has atleast two senses and which have occurred at least twice in the SENSEVAL-2 dataset are chosen

Token	Senses	Token	Senses
church	4	individual	2
field	13	child	4
bell	10	risk	4
rope	2	eye	5
band	12	research	2
ringer	4	team	2
tower	3	version	6
group	3	copy	3
year	4	loss	8
vicar	3	colon	5
sort	4	leader	2
country	5	discovery	4
woman	4	education	6
cancer	5	performance	5
cell	7	school	7
type	6	pupil	3
growth	6	student	2

[B.1] MFS Results on selected words

Word Vectors	Accuracy
SkipGram-BNC	63.63
SkipGram-Brown	48.38
SkipGram-Google-News	60.6
Senna	57.57
Glove	**66.66**
Global Context	51.51
Metaoptimize	27.27
RNN	51.51
Multilingual	63.4

Table: English WSD results for selected words from SENSEVAL-2 dataset

Experiments

A. Experiments on WSD
   1. Experiments on WSD using Skip-Gram model
      - Hindi (Newspaper)
      - English (SENSEVAL-2 and SENSEVAL-3)
   2. Experiments on WSD using different word vector models
   3. Comparing WSD results using different sense vector models
      - Retrofitting Sense Vector Model (English)
   4. Experiments on WSD for words which do not exists in SemCor

B. Experiments on selected words (34 polysemous words from SENSEVAL-2 corpus)
   1. Experiments using different word vector models
   2. Comparing results with various sizes of vector dimensions

[B.2] Comparing MFS results with various sizes of vector dimensions

Word Vectors	Accuracy
SkipGram-BNC-1500	60.61
SkipGram-BNC-1000	60.61
SkipGram-BNC-500	66.67
SkipGram-BNC-400	**69.69**
SkipGram-BNC-300	63.64
SkipGram-BNC-200	60.61
SkipGram-BNC-100	48.49
SkipGram-BNC-50	51.52

MFS for Indian Languages

- *Polyglot* word embeddings are used for obtaining MFS.
  - word embeddings are trained using Wikipedia data.

- Currently, system is working for *Marathi*, *Bengali*,*Gujarati*, *Sanskrit*, *Assamese*, *Bodo*, *Oriya*, *Kannada*, *Tamil*, *Telugu*, *Malayalam* and *Punjabi*.

- Due to lack of gold data, we could not evaluate results

- APIs are developed for finding the MFS for a word

Conclusion

- An unsupervised approach is designed for finding the MFS by using word embeddings.
- Tested MFS results on WSD and some selected words.
- Performance is compared with different word vector models and various size of the dimensions.
- Our sense vector model always show better results on nouns, verbs and adverbs as compared to *retrofitting* model.
- Approach can be easily ported to various domains and across languages.
- APIs are created for detecting the MFS for English and Indian languages.

Future Work

- Domain Specific MFS evaluation
- Evaluation on more languages
- Evaluation of MFS of *tatsama* words on closely related family of languages
- Try different heuristics sense embeddings creation
- Use different sense repositories like Universal WordNet
- Automatic synset rankings can be done using the same approach with mixed-domain corpora

Publications

- Sudha Bhingardive, Samiulla Shaikh and Pushpak Bhattacharyya. “Neighbors Help: Bilingual Unsupervised WSD Using Context”, In proceedings of The 51st Annual Meeting of the Association for Computational Linguistics (**ACL 2013**), Sofia, Bulgaria.

- Sudha Bhingardive, Tanuja Ajotikar, Irawati Kulkarni, Malhar Kulkarni and Pushpak Bhattacharyya,. “Semi-Automatic Extension of Sanskrit Wordnet using Bilingual Dictionary”, Global WordNet Conference, (**GWC 2014**), Tartu, Estonia, 25-29 January, 2014.

- Sudha Bhingardive, Ratish Puduppully , Dhirendra Singh and Pushpak Bhattacharyya. “Merging Senses of Hindi WordNet using Word Embeddings”, International Conference on Natural Language Processing, (**ICON 2014**), Goa,India.

- Sudha Bhingardive, Dhirendra Singh, Rudramurty V, Hanumnat Redkar and Pushpak Bhattacharyya, “Unsupervised Most Frequent Sense Detection using Word Embeddings”, North American Chapter of the Association for Computational Linguistics – Human Language Technologies (**NAACL HLT 2015**) , Denver, Colorado, USA.

Publications

- Devendra Singh Chaplot, Sudha Bhingardive and Pushpak Bhattacharyya. “IndoWordnet Visualizer: A Graphical User Interface for Browsing and Exploring Wordnets of Indian Languages.”, Global WordNet Conference, (**GWC 2014**), Tartu, Estonia, 25-29 January, 2014.

- Hanumant Redkar, Sudha Bhingardive , Diptesh Kanojia and Pushpak Bhattacharyya, “WorldWordNet Database Structure: An Efficient Schema for Storing Information of WordNets of the World”, (**AAAI-2015)**, Austin, USA.

- Dhirendra Singh, Sudha Bhingardive, Kevin Patel and Pushpak Bhattacharyya, “Using Word Embeddings and WordNet features for MultiWord Expression Extraction”, Linguistic Society of India (**LSI 2015),** JNU, Delhi, India.

- Dhirendra Singh, Sudha Bhingardive and Pushpak Bhattacharyya, “Detection of Light Verb Constructions Using Word Embeddings and WordNet based features ”, International Conference on Natural Language Processing, (**ICON 2015**), India

Publications

- Sudha Bhingardive, Dhirendra Singh, Rudramurthy R and Pushpak Bhattacharyya. “Using Word Embeddings for Bilingual Unsupervised WSD”, International Conference on Natural Language Processing, (**ICON 2015**), India.

- Sudha Bhingardive, Hanumant Redkar, Prateek Sappadla, Dhirendra Singh and Pushpak Bhattacharyya. “IndoWordNet-based Semantic Similarity Measurement”, Global WordNet Conference, (**GWC 2016**), Romania, 2016.

- Hanrpreet Arora, Sudha Bhingardive, and Pushpak Bhattacharyya, “Most Frequent Sense Detection Using BableNet”, Global WordNet Conference (**GWC 2016**), Romania, 2016.

- Dhirendra Singh, Sudha Bhingardive and Pushpak Bhattacharyya, “Detection of Light Verb Constructions Using WordNet “, Global WordNet Conference, (**GWC 2016**), Romania, 2016.

- “Synset Ranking of Hindi WordNet” (submitted to **LREC 2016**)

Tutorial

- Sudha Bhingardive, Rudramurty V, Kevin Patel, Prerana Singhal, “Deep Learning and Distributed Word Representations”, **ICON 2015**. (**Tutorial**)

References

- Harris, Z. S. 1954. “Distributional structure.” Word, 10:146–162.
- Tomas Mikolov, Chen Kai, Corrado Greg and Dean Jeffrey. 2013. “Efficient Estimation of Word Representations in Vector Space”, In Proceedings of Workshop at ICLR, 2013.
- Patrick Pantel and Dekang Lin. 2002. “Discovering word senses from text.” In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '02). ACM, New York, NY, USA.
- McCarthy, D., Koeling, R., Weeds, J., & Carroll, J. 2004. “Using automatically acquired predominant senses for word sense disambiguation”. In Proceedings of the ACL.
- Agirre, E. and Edmonds, P. 2007. “Word Sense Disambiguation: Algorithms and Applications”. Springer Publishing Company, Incorporated, 1st edition.
- Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. 2003. “A neural probabilistic language model”. J. Mach. Learn. Res., 3:1137–1155.
- Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. 2011. “Natural language processing (almost) from scratch”. The Journal of Machine Learning Research, 12:2493–2537.
- Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. “Improving word representations via global context and multiple word prototypes”. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, USA, 873-882.

References

- Buitelaar, Paul, and Bogdan Sacaleanu. 2001. “Ranking and selecting synsets by domain relevance”. Proceedings of WordNet and Other Lexical Resources: Applications, Extensions and Customizations, NAACL 2001 Workshop.
- Mohammad, Saif, and Graeme Hirst. 2006. “Determining Word Sense Dominance Using a Thesaurus”. EACL.
- Lapata, Mirella, and Chris Brew. 2004. “Verb class disambiguation using informative priors”. Computational Linguistics 30.1 (2004): 45-73.
- O. Bojar, V. Diatka, P. Rychlý, P. Stranák, V. Suchomel, A. Tamchyna, and D. Zeman. 2014. “HindEnCorp-Hindi-English and Hindi-only Corpus for Machine Translation”. In Proceedings of LREC. 2014, 3550-3555.
- Diana Mccarthy, Rob Koeling, Julie Weeds, and John Carroll. 2004. “Finding predominant word senses in untagged text”. In In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, pages 280–287.
- Xinxiong Chen, Zhiyuan Liu and Maosong Sun. 2014. “A Unified Model for Word Sense Representation and Disambiguation”, Proceedings of ACL 2014.
- Tang, D.,Wei, F., Yang, N., Zhou, M., Liu, T., and Qin, B. 2014. “Learning sentiment-specific word embedding for twitter sentiment classification”. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 1555–1565.

References

- Manaal Faruqui and Chris Dyer. 2014. “Community Evaluation and Exchange of Word Vector”,s at wordvectors.org ,Proceedings of System Demonstrations, ACL 2014
- Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukar Burget, and Jan Honza Cernocky. 2011. “RNNLM - Recurrent Neural Network Language Modeling Toolkit”, In: ASRU 2011
- Jauhar, Sujay Kumar, Chris Dyer, and Eduard Hovy. 2015. "Ontologically Grounded Multi-sense Representation Learning for Semantic Vector Space Models.”, ACL 2015.

Thank You!!!

Extra slides

Evaluating the quality of Hindi Word Vectors

- We created a similarity word pair dataset by translating the standard similarity word pair dataset (Agirre et al., 2009) available for English.

- Three annotators were instructed to give the score for each word-pair based on the semantic similarity and relatedness.

- The scale was chosen between 0 - 10.

- Average inter-annotator agreement = 0.73

Why Word Embeddings?

- Consider the *one hot* representation for words “song” and “music”

  [“song”] = [1 0 0]

  [“music”] = [0 1 0]

  [“box”] = [0 0 1]

- similarity (“song” , “music”) = ?

- In general, we can not capture the similarity between any two words using *one hot* representation

Distributional Hypothesis

- *Similar words occur in similar context* (Harris, 1954)

- Consider following example,

I ate "X" in the restaurant.

"X" was very spicy.

I like to eat "X" with only chopsticks.

- What is "*X*" ?

Distributional Hypothesis contd..

- *Similar words occur in similar context* (Harris, 1954)

- Consider following example,

I ate "X" in the restaurant.

"X" was very spicy.

I like to eat "X" with only chopsticks.

- What is "*X* " ?
  - A food item

Distributional Hypothesis contd..

- *Similar words occur in similar context* (Harris, 1954)

- Consider following example,

I ate "X" in the restaurant.

"X" was very spicy.

I like to eat "X" with only chopsticks.

- What is "*X* " ?
  - A food item
- How humans recognized what word "*X* " could be ?
  - looking at the context in which "*X* " appears

{*"ate", "restaurant", "very spicy", "eat", "chopsticks"}*

Distributional Hypothesis contd..

- *Similar words occur in similar context* (Harris, 1954)

- Consider following example,

I ate "X" in the restaurant.

"X" was very spicy.

I like to eat "X" with only chopsticks.

- What is "*X* " ?
  - A food item
- How humans recognized what word "*X* " could be ?
  - looking at the context in which "*X* " appears

{*"ate", "restaurant", "very spicy", "eat", "chopsticks"}*

- What is "Y" in "*Y was not that spicy*"

Distributional Hypothesis contd..

- Co-occurrence matrix

	X	*Y*	*ate*	*restaurant*	*kitchen*	*sweet*	*spicy*	*chopsticks*	*spoon*	*drink*
X	*0*	*0*	*1*	*1*	*0*	*0*	*1*	*1*	*0*	*0*
Y	*0*	*0*	*1*	*0*	*1*	*0*	*1*	*0*	*1*	*1*
ate	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*1*	*0*	*0*
.										
.										
drink	*0*	*1*	*0*	*0*	*1*	*1*	*0*	*0*	*1*	*0*

X and *Y* are represented as,

$X = [0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0]$ $Y = [0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1]$