# Domain Driven Decompositional Semantics

**Pranjal Singh**
Supervisor: Dr. Amitabha Mukerjee

B.Tech - M.Tech Dual Degree

Thesis Defense

Department of Computer Science & Engineering
IIT Kanpur


June 15, 2015

# Outline

# Outline

# Introduction to Decompositional Semantics

Decompositional Semantics is a way to describe a language entity word/paragraph/document by a constrained representation that identifies the most relevant representation conveying the semantics of the whole.

For example, a document can be broken into aspects such as its tf-idf representation, distributed semantics vector, etc.

# Introduction to Decompositional Semantics

**Why need Decompositional Semantics?**

- It is language independent
- It decomposes language entity into various aspects that are latent in its meaning
- All aspects are important in their own ways

# Introduction to Decompositional Semantics

Decompositional Semantics in Sentiment Analysis domain,

- A set of documents $D = \{d_1, \ldots, d_{|D|}\}$
- A set of aspects $A = \{a_1, \ldots, a_{|M|}\}$
- Training data for $n$ ($n < |D|$) documents, $\mathcal{T} = \{l_{d_1}, \ldots, l_{d_n}\}$

Example :

| Documents | tf-idf | Word Vector Average | Document Vector | BOW |
|---|---|---|---|---|
| $d_1$ | 0 | 0 | 1 | 0 |
| $d_2$ | 0 | 1 | 1 | 0 |
| $d_3$ | 1 | 0 | 0 | 1 |
| $d_4$ | × | × | × | × |
| $d_5$ | 1 | 1 | 1 | 1 |

Using $\mathcal{T}$, $D$ and $A$, the supervised classifier $\mathcal{C}$ learns a representation to predict sentiments of individual documents.

# Problem Statement

## Better Language Representation

- To highlight the vitality of Decompositional Semantics in language representation
- To use Distributional Semantics for under resourced languages such as Hindi
- To demonstrate the effect of various parameters on language representation

# Contribution of this thesis

## Hindi

- Better representation of Hindi text using Distributional semantics
- Achieved state-of-the-art results for sentiment analysis on product and movie review corpus

Paper accepted in regICON'15

## New Corpus

- Released a corpus of 700 Hindi movie reviews
- Largest corpus in Hindi in reviews domain

## English

- Proposed a more generic representation of English text
- Achieved state-of-the-art results for sentiment analysis on IMDB movie reviews and Amazon electronics reviews

Submitted in EMNLP'15

# Contribution of this thesis

## Hindi

- Better representation of Hindi text using Distributional semantics
- Achieved state-of-the-art results for sentiment analysis on product and movie review corpus

Paper accepted in regICON'15

## New Corpus

- Released a corpus of 700 Hindi movie reviews
- Largest corpus in Hindi in reviews domain

## English

- Proposed a more generic representation of English text
- Achieved state-of-the-art results for sentiment analysis on IMDB movie reviews and Amazon electronics reviews

Submitted in EMNLP'15

# Contribution of this thesis

## Hindi

- Better representation of Hindi text using Distributional semantics
- Achieved state-of-the-art results for sentiment analysis on product and movie review corpus

Paper accepted in regICON'15

## New Corpus

- Released a corpus of 700 Hindi movie reviews
- Largest corpus in Hindi in reviews domain

## English

- Proposed a more generic representation of English text
- Achieved state-of-the-art results for sentiment analysis on IMDB movie reviews and Amazon electronics reviews

Submitted in EMNLP'15

# Outline

# Background on Language Representation

## Bag of Words(BOW) Model

- Document $d_i$ represented by $v_{d_i} \in \mathbb{R}^{|V|}$
- Each element in $v_{d_i}$ denotes presence/absence of each word
- **Drawbacks:**
  - High-dimensionality
  - Ignores word ordering
  - Ignores word context
  - Very sparse
  - No relative importance to words

# Background on Language Representation

## Bag of Words(BOW) Model

- Document $d_i$ represented by $v_{d_i} \in \mathbb{R}^{|V|}$
- Each element in $v_{d_i}$ denotes presence/absence of each word
- **Drawbacks**:
  - High-dimensionality
  - Ignores word ordering
  - Ignores word context
  - Very sparse
  - No relative importance to words

# Background on Language Representation

## Term Frequency-Inverse Document Frequency(tf-idf) Model

- Document $d_i$ represented by $v_{d_i} \in \mathbb{R}^{|V|}$
- Each element in $v_{d_i}$ is the product of term frequency and inverse document frequency: $tfidf(t, d) = tf(t, d) \times \log(\frac{\|D\|}{df(t)})$
- Gives weights to terms which are less frequent and hence important
- **Drawbacks**:
  - High-dimensionality
  - Ignores word ordering
  - Ignores word context
  - Very sparse
  - No relative importance to words

# Background on Language Representation

## Term Frequency-Inverse Document Frequency(tf-idf) Model

- Document $d_i$ represented by $v_{d_i} \in \mathbb{R}^{|V|}$
- Each element in $v_{d_i}$ is the product of term frequency and inverse document frequency: $tfidf(t, d) = tf(t, d) \times \log(\frac{\|D\|}{df(t)})$
- Gives weights to terms which are less frequent and hence important
- **Drawbacks**:
  - High-dimensionality
  - Ignores word ordering
  - Ignores word context
  - Very sparse
  - ~~No relative importance to words~~

# Background on Language Representation

**Distributed Representation of Words(Mikolov et al., 2013b)**

- Each word $w_i \in V$ is represented using a vector $v_{w_i} \in \mathbb{R}^k$
- The vocabulary $V$ can be represented by a matrix $V \in \mathbb{R}^{k \times |V|}$
- Vectors ($v_{w_i}$) should encode the semantics of the words in vocabulary
- **Drawbacks**:
  - Ignores exact word ordering
  - Cannot represent documents as vectors without composition

# Background on Language Representation

**Distributed Representation of Words(Mikolov et al., 2013b)**

- Each word $w_i \in V$ is represented using a vector $v_{w_i} \in \mathbb{R}^k$
- The vocabulary $V$ can be represented by a matrix $V \in \mathbb{R}^{k \times |V|}$
- Vectors ($v_{w_i}$) should encode the semantics of the words in vocabulary
- **Drawbacks**:
  - Ignores exact word ordering
  - Cannot represent documents as vectors without composition

# Background on Language Representation

**Distributed Representation of Documents(Le and Mikolov, 2014)**

- Each document $d_i \in D$ is represented using a vector $v_{d_i} \in \mathbb{R}^k$
- The set $D$ can be represented by a matrix $D \in \mathbb{R}^{k \times |D|}$
- Vectors ($v_{d_i}$) should encode the semantics of the documents
- **Comments**:
  - Can represent documents
  - Ignores contribution of indvidual word while building document vectors

# Background on Language Representation

**Distributed Representation of Documents(Le and Mikolov, 2014)**

- Each document $d_i \in D$ is represented using a vector $v_{d_i} \in \mathbb{R}^k$
- The set $D$ can be represented by a matrix $D \in \mathbb{R}^{k \times |D|}$
- Vectors $(v_{d_i})$ should encode the semantics of the documents
- **Comments**:
  - Can represent documents
  - Ignores contribution of indvidual word while building document vectors

# Background on Sentiment Analysis

- Pang et al.(2004) obtained 87.2% accuracy on a dataset that discarded objective sentences and used text categorization techniques on the subjective sentences
- Socher et al.(2013) used recursive neural network over sentiment treebank for sentiment classification
- Le and Mikolov (2014) use document vector model and obtained 92.6% accuracy on IMDB movie review dataset

# Background on Sentiment Analysis

There has been limited work on sentiment analysis in Hindi

- Joshi et al.(2010) used In-language sentiment analysis, Machine Translation and Resource Based Sentiment Analysis to achieve 78.1% accuracy
- Mukherjee et al.(2012) presented the inclusion of discourse markers in a BOW model to improve the sentiment classification accuracy by 2-4%
- Mittal et al.(2013) incorporate hand-coded rules dealing with negation and discourse relations achieving 80.2% accuracy

# Background on Sentiment Analysis

There has been limited work on sentiment analysis in Hindi

- Joshi et al.(2010) used In-language sentiment analysis, Machine Translation and Resource Based Sentiment Analysis to achieve 78.1% accuracy
- Mukherjee et al.(2012) presented the inclusion of discourse markers in a BOW model to improve the sentiment classification accuracy by 2-4%
- Mittal et al.(2013) incorporate hand-coded rules dealing with negation and discourse relations achieving 80.2% accuracy

# Outline

# Hindi Product and Movie Review Corpus

- Product Review dataset (LTG, IIIT Hyderabad) contains 350 Positive reviews and 350 Negative reviews
- Movie Review dataset (CFILT, IIT Bombay) contains 127 Positive reviews and 125 Negative reviews
- Each review is around 1-2 sentences long and the sentences are mainly focused on sentiment, either positive or negative.

# 700-Movie Review Corpus

- We collected Hindi movie reviews from websites such as Dainik Jagran and Navbharat Times
- The movie reviews are longer than the previous corpus and contains subjects other than sentiment

| **Overall** | |
|---|---|
| Positive Reviews | 356 |
| Negative Reviews | 341 |
| Total Reviews | 697 |
| 29.7 sentences per document | |
| 494.6 words per document | |

Table 1 : Statistics of Movie Reviews from Jagran and Navbharat

# English Corpus

## IMDB Movie Reviews

- Contains 25,000 positive and 25,000 negative reviews for training purpose
- It also contains an additional 50,000 unlabeled documents for unsupervised learning

## Amazon Product Reviews

- There are 3 review datasets: Watches, Electronics and MP3 each of size 30.8MB, 728.4MB and 27.7MB respectively
- Electronics dataset consists of 1,241,778 reviews, Watches Dataset consists of 68,356 reviews and MP3 Dataset consists of 31,000 reviews
- Datasets are split into 80-20 ratio for training and testing

# Wikipedia

## Hindi

Hindi Wikipedia text dump (approx. 290MB) containing around 24M words with 724K words in the vocabulary.

## English

English Wikipedia text dump (approx. 20.3GB) contains around 3.5B words with 7.8M words in the vocabulary.

# Outline

# Distributed Word Representation

## Skipgram

- Each current word acts as an input to a log-linear classifier with continuous projection layer, and predict words within a certain range before and after the current word
- The objective is to maximize the probability of the context given a word:

$$p(c|w;\theta) = \frac{\exp^{v_c \cdot v_w}}{\sum_{c' \in C} \exp^{v_c \cdot v_w}}$$

- $v_c$ and $v_w \in R^d$ are vector representations for context $c$ and word $w$ respectively. $C$ is the set of all available contexts. The parameters $\theta$ are $v_{c_i}$, $v_{w_i}$ for $w \in V$, $c \in C$, $i \in 1, ...., d$

# Distributed Word Representation

- Weights between the input layer and the output layer can be represented by a $V \times N$ matrix $\mathbf{W}$
- Each row of $\mathbf{W}$ is the $N$-dimension vector representation $v_w$ of the associated word of the input layer
- Given a word, assuming $x_k = 1$ and $x_{k'} = 0$ for $k' \neq k$, then

$$h = x^T W = W_{(k,.)} := v_{w_I}$$
$$u_j = {v'}_{w_j}^T . h$$

- $v_{w_I}$ is the vector representation of the input word $w_I$ and $u_j$ is the score of each word in the vocabulary
- There is a different weight matrix $\mathbf{W'} = \{w'_{ij}\}$ which is a $N \times V$ matrix between hidden and output layer
- Softmax function is used to predict probabilities and Stochastic Gradient Descent is used to update the parameters of the model

# Distributed Document Representation

## Motivation

- Drawbacks in BOW like sparsity, high-dimensionality, inability to encode context information and consider word ordering
- Composition models alone cannot represent documents (Blacoe and Lapata, 2012)
- Recursive Tensor Neural Networks (Socher et al.,2013) are computationally expensive and cannot be composed into document vectors when there are multiple sentences due to parsing issues
- Presence of similarity measures to deal with synonyms or semantically similar documents

# Distributed Document Representation

- Every document is now mapped to a unique vector and id, represented by a matrix $D$
- Word vector matrix $W$ is shared across all documents and contexts are now separately sampled for each document
- The only difference in this model is that $h$ is now constructed with both $W$ and $D$.

# Semantic Composition

The Principle of Compositionality is that meaning of a complex expression is determined by the meaning of its constituents and the rules which guide this combination. It is also known as Frege's Principle.
For example,

The movie is funny and the screenplay is good

In the above sentence, consider the word vectors are represented by $w(x)$ and the sentence vector as $S(x)$. Hence,

$$S(x) = c_1 w_1(x) \Theta c_2 w_2(x) \Theta c_3 w_3(x) \Theta c_4 w_4(x) \ldots \Theta c_k w_k(x) \qquad (1)$$

where $\Theta$ can be any operation(e.g., addition, multiplication) and $c_i$s are constants.

# Semantic Composition

- We describe two approaches to incorporate graded weighting into word vectors for building document vectors.
- Let $v_{w_i}$ be the vector representation of the $i^{th}$ word. Then document vector $v_{d_i}$ for $i^{th}$ document is:

$$v_{d_i} = \begin{cases} 0 & w_k \in stopwords \\ \sum_{w_k \in d_i} v_{w_k} & w_k \notin stopwords \end{cases}$$

The above equation is 0-1 step-function which ignores contribution of all stop words.

- Another schema which incorporates idf weight is:

$$v_{d_i} = \begin{cases} 0 & idf(w_k, d_i) \leq \delta \\ \sum_{w_k \in d_i} idf(w_k, d_i).v_{w_k} & otherwise \end{cases}$$

where $\delta$ is a pre-defined threshold below which the word has no importance and above which the idf terms gives importance to that particular word.

# Semantic Composition

- We describe two approaches to incorporate graded weighting into word vectors for building document vectors.
- Let $v_{w_i}$ be the vector representation of the $i^{th}$ word. Then document vector $v_{d_i}$ for $i^{th}$ document is:

$$v_{d_i} = \begin{cases} 0 & w_k \in stopwords \\ \sum_{w_k \in d_i} v_{w_k} & w_k \notin stopwords \end{cases}$$

The above equation is 0-1 step-function which ignores contribution of all stop words.

- Another schema which incorporates idf weight is:

$$v_{d_i} = \begin{cases} 0 & idf(w_k, d_i) \leq \delta \\ \sum_{w_k \in d_i} idf(w_k, d_i).v_{w_k} & otherwise \end{cases}$$

where $\delta$ is a pre-defined threshold below which the word has no importance and above which the idf terms gives importance to that particular word.

# Semantic Composition

- We describe two approaches to incorporate graded weighting into word vectors for building document vectors.
- Let $v_{w_i}$ be the vector representation of the $i^{th}$ word. Then document vector $v_{d_i}$ for $i^{th}$ document is:

$$v_{d_i} = \begin{cases} 0 & w_k \in stopwords \\ \sum_{w_k \in d_i} v_{w_k} & w_k \notin stopwords \end{cases}$$

The above equation is 0-1 step-function which ignores contribution of all stop words.

- Another schema which incorporates idf weight is:

$$v_{d_i} = \begin{cases} 0 & idf(w_k, d_i) \leq \delta \\ \sum_{w_k \in d_i} idf(w_k, d_i).v_{w_k} & otherwise \end{cases}$$

where $\delta$ is a pre-defined threshold below which the word has no importance and above which the idf terms gives importance to that particular word.

# Semantic Composition

| Composition | Accuracy |
|---|---|
| Multiplication | 50.30 |
| Average | 88.42 |
| Weighted Average | **89.56** |

Table 2 : Results of Vector Composition with different Operations

| **Method** | **Weight** | **Accuracy(1)** | **Accuracy(2)** |
|---|---|---|---|
| 0-1 Weighting | 0 | 93.84 | 93.06 |
| | 1 | **93.91** | **93.18** |
| Graded idf Weighting | 2 | **93.89** | 93.17 |
| | 2.5 | 93.87 | 93.16 |
| | 2.8 | 93.86 | 93.16 |
| | 3 | 93.86 | **93.22** |
| | 4 | 93.83 | 93.12 |

Table 3 : Results on IMDB Movie Reviews(Composite Document Vector);Accuracy(2) is when we exclude tf-idf features

# Effect of Context Size

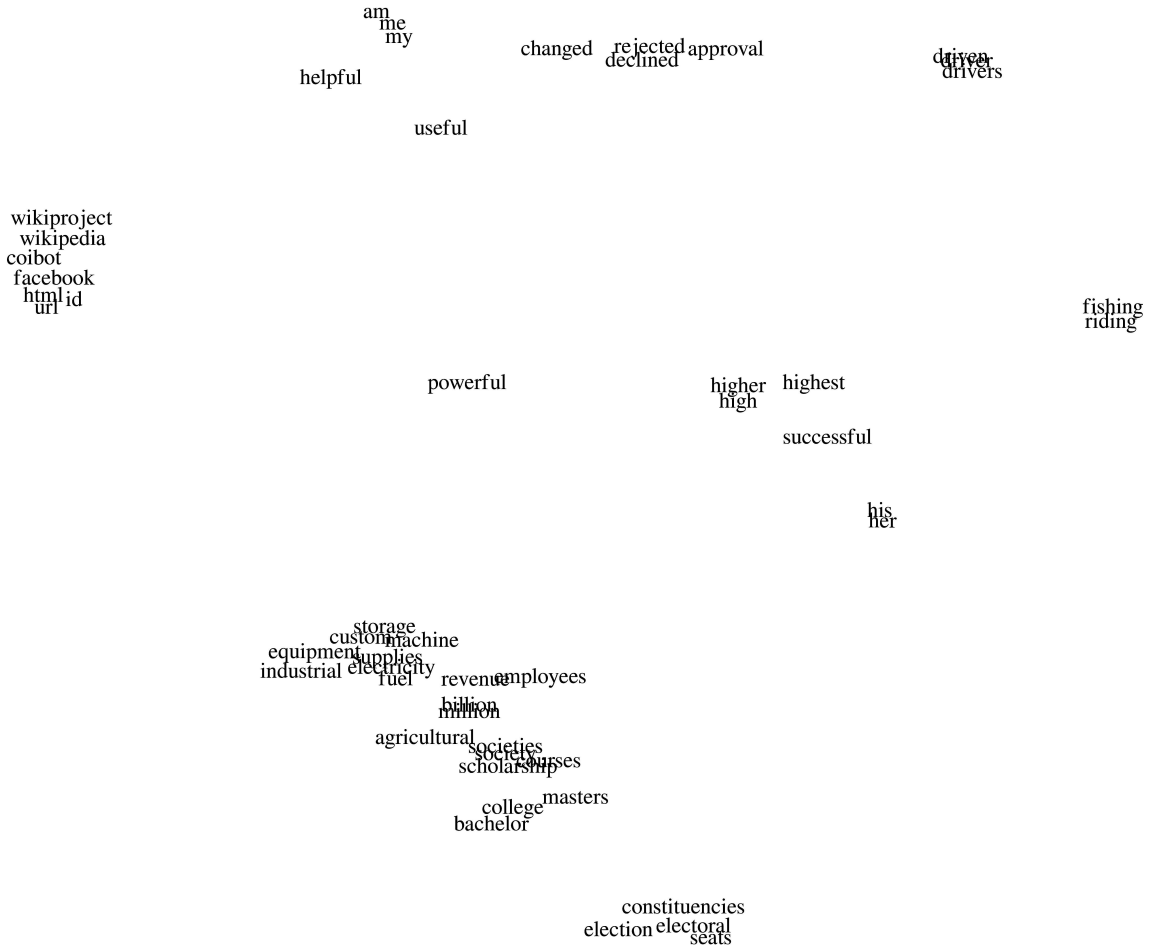


Figure 1 : High Dimensional representation of Wiki Text with Context Size 5

# Effect of Context Size

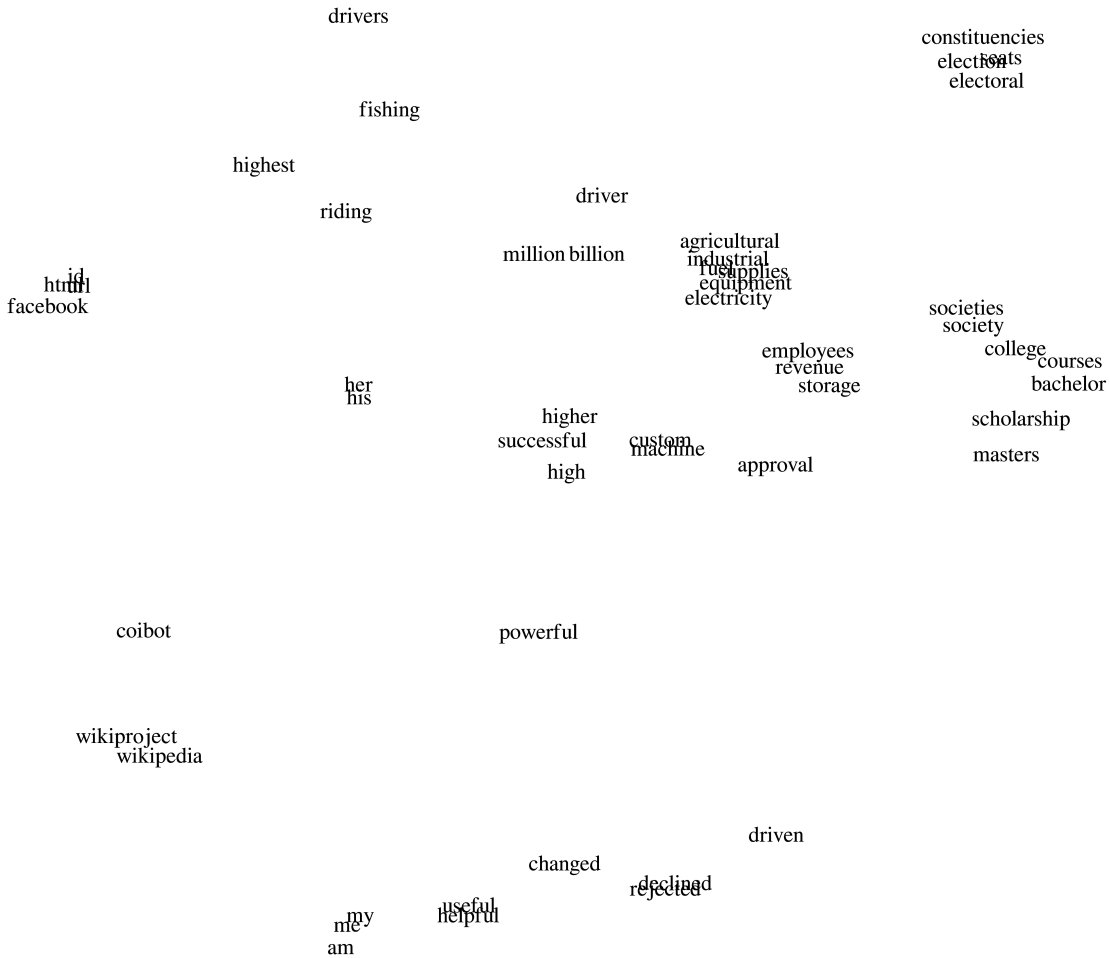


Figure 2 : High Dimensional representation of Wiki Text with Context Size 10

# Effect of Context Size

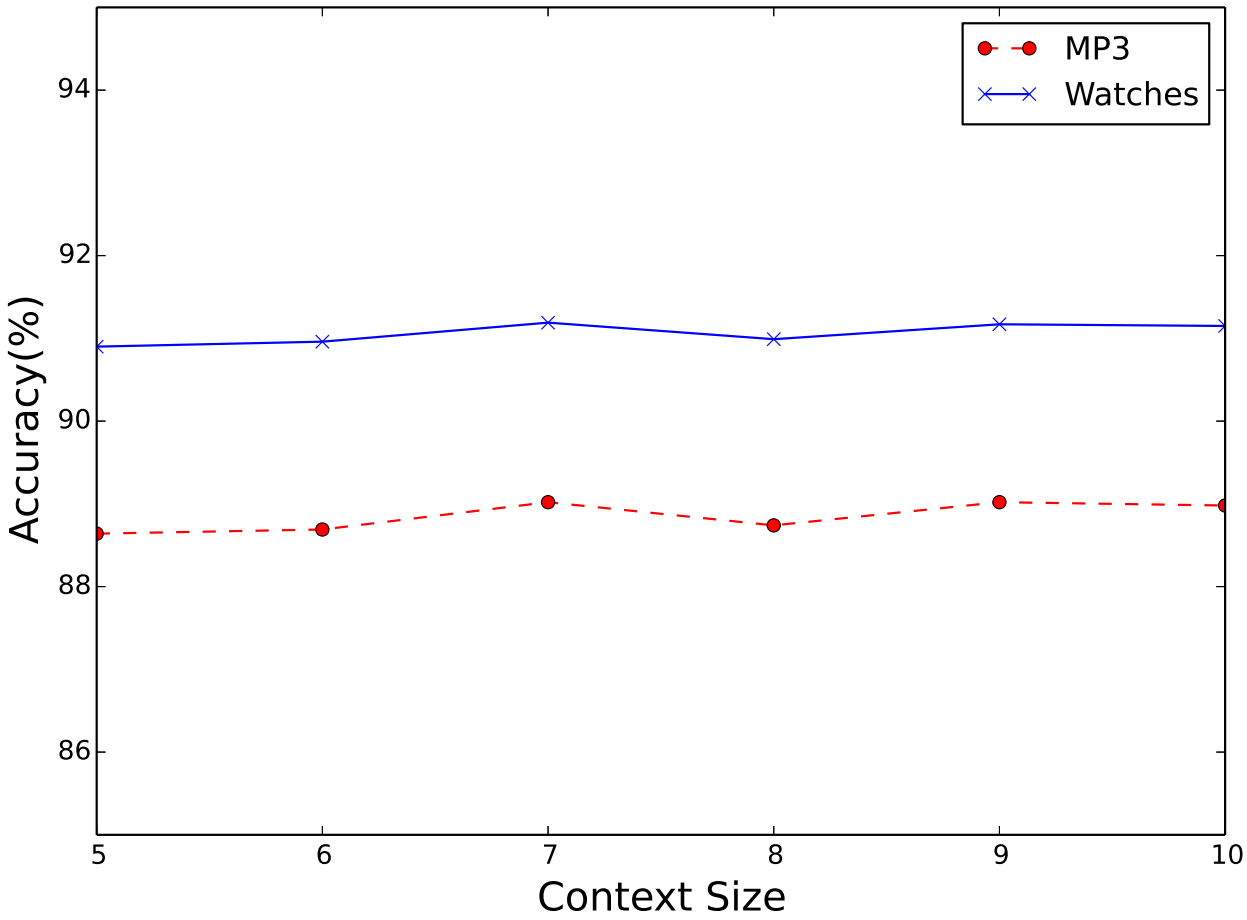


Figure 3 : Variation of Accuracy with Different Context Size on Watches and MP3 Datasets

# SkipGram or CBOW?

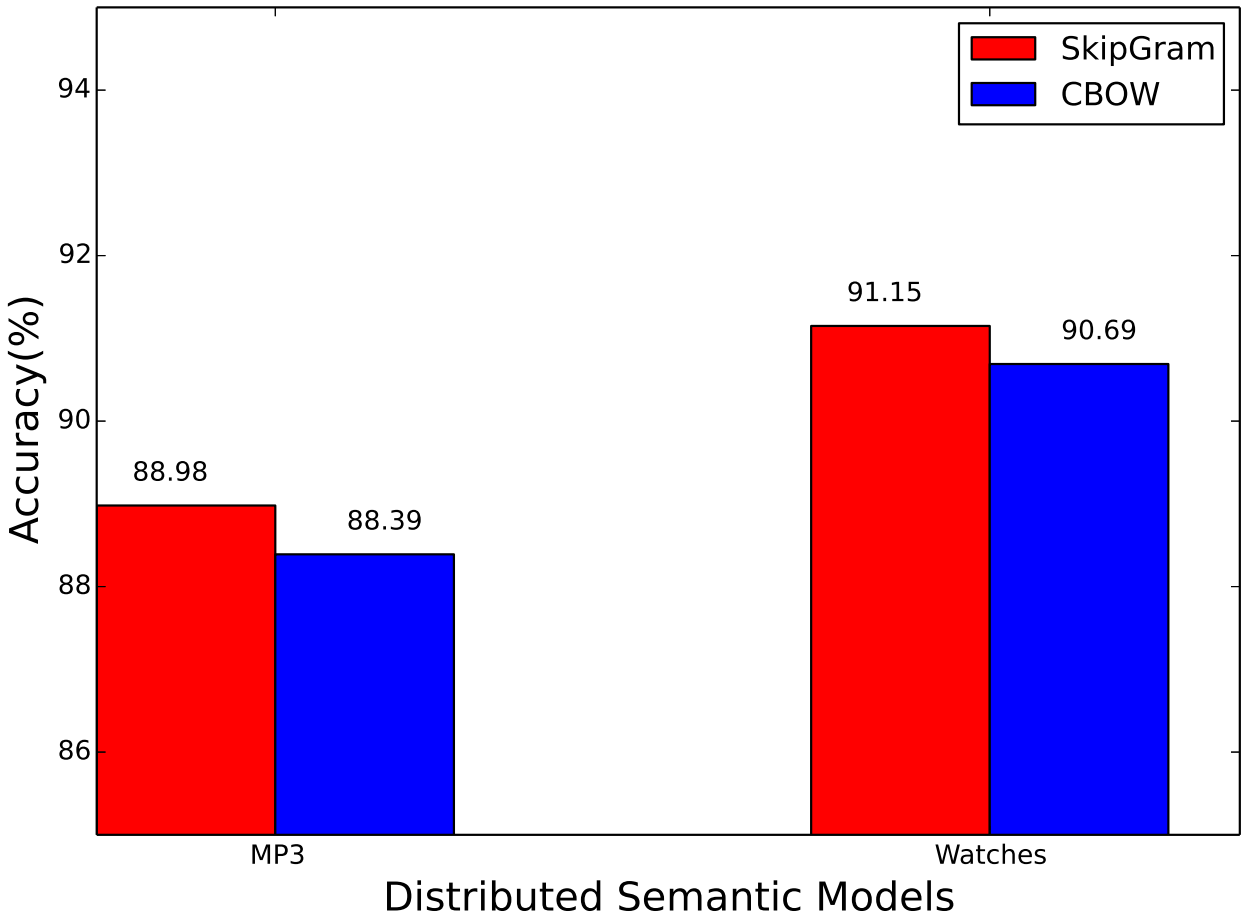


Figure 4 : Variation of Accuracy with skipgram and cbow on Watches and MP3 Datasets.

# Work Flow

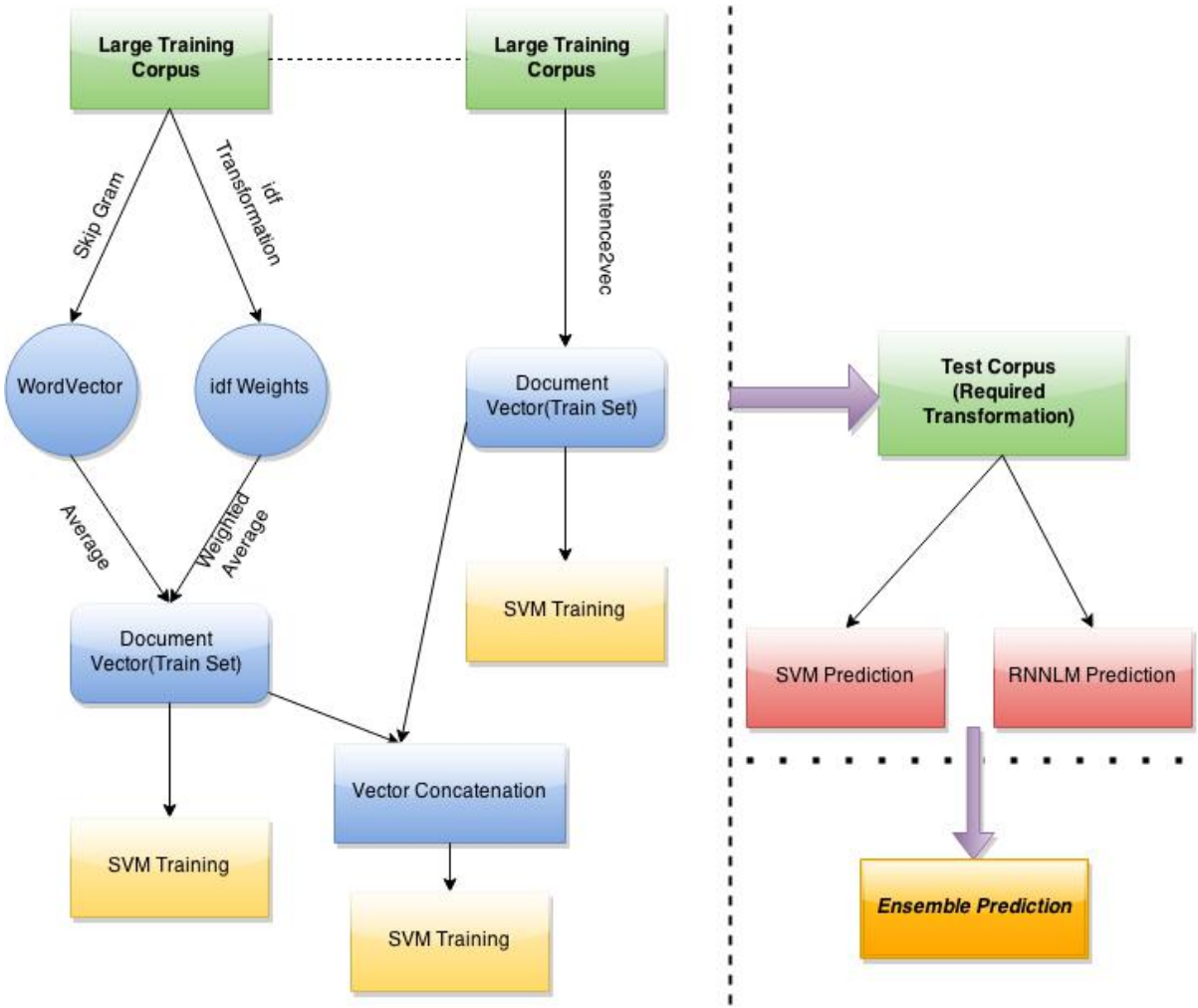


Figure 5 : Work Flow

# Outline

# Result on English Dataset

| Method | IMDB | Amazon | Hindi |
|---|---|---|---|
| RNNLM (Baseline) | 86.45 | 90.03 | 78.84 |
| Paragraph Vector(Le and Mikolov,2014) | 92.58 | 91.30 | 74.57 |
| Averaged Vector | 88.42 | 88.52 | 79.62 |
| Weighted Average Vector | 89.56 | 88.63 | 85.90 |
| Composite Document Vector | 93.91 | 92.17 | 90.30 |

Table 4 : Comparison of accuracies on 3 Datasets(IMDB, Amazon Electronics Review and Hindi Movie Reviews(IITB)) for various types of document composition models. The state of the art for these tasks are: IMDB: 92.58%; Amazon:85.90%, Hindi:79.0%.

# Result on English Dataset

| Method | Accuracy |
|---|---|
| Maas et al.(2011) | 88.89 |
| NBSVM-bi (Wang & Manning, 2012) | 91.22 |
| NBSVM-uni (Wang & Manning, 2012) | 88.29 |
| SVM-uni (Wang & Manning, 2012) | 89.16 |
| Paragraph Vector (Le and Mikolov(2014)) | 92.58 |
| WordVector+Wiki(Our Method) | 88.60 |
| Weigted WordVector+Tfldf(Our Method) | 89.56 |
| Weighted WordVector+Tfldf+Document Vector | **93.91** |
| Ensemble of Enhanced Document Vector and RNNLM | 94.19 |

Table 5 : Results on IMDB Movie Review Dataset

# Result on English Dataset

| Method | Accuracy |
|---|---|
| WordVector Averaging | 88.42 |
| Weighted WordVector Average | 89.56 |
| Weigted WordVector Averaging+Wiki | 88.60 |
| Weigted WordVector Averaging+Tfldf | 90.67 |
| WordVector Averaging+Document Vector | 93.18 |
| WordVector Averaging+Wiki+Document Vector | 93.18 |
| WordVector Averaging+Document Vector+RNNLM | 93.70 |
| WordVector Averaging+Wiki+Document Vector+RNNLM | 93.57 |
| WordVector Averaging+Tfldf+Document Vector | 93.91 |
| WordVector Averaging+Wiki+Document Vector+Tfldf | 93.55 |
| WordVector Averaging+Tfldf+Document Vector+RNNLM | **94.19** |

Table 6 : Comparison of results on IMDB Movie Review Dataset with Various Features

# Result on English Dataset

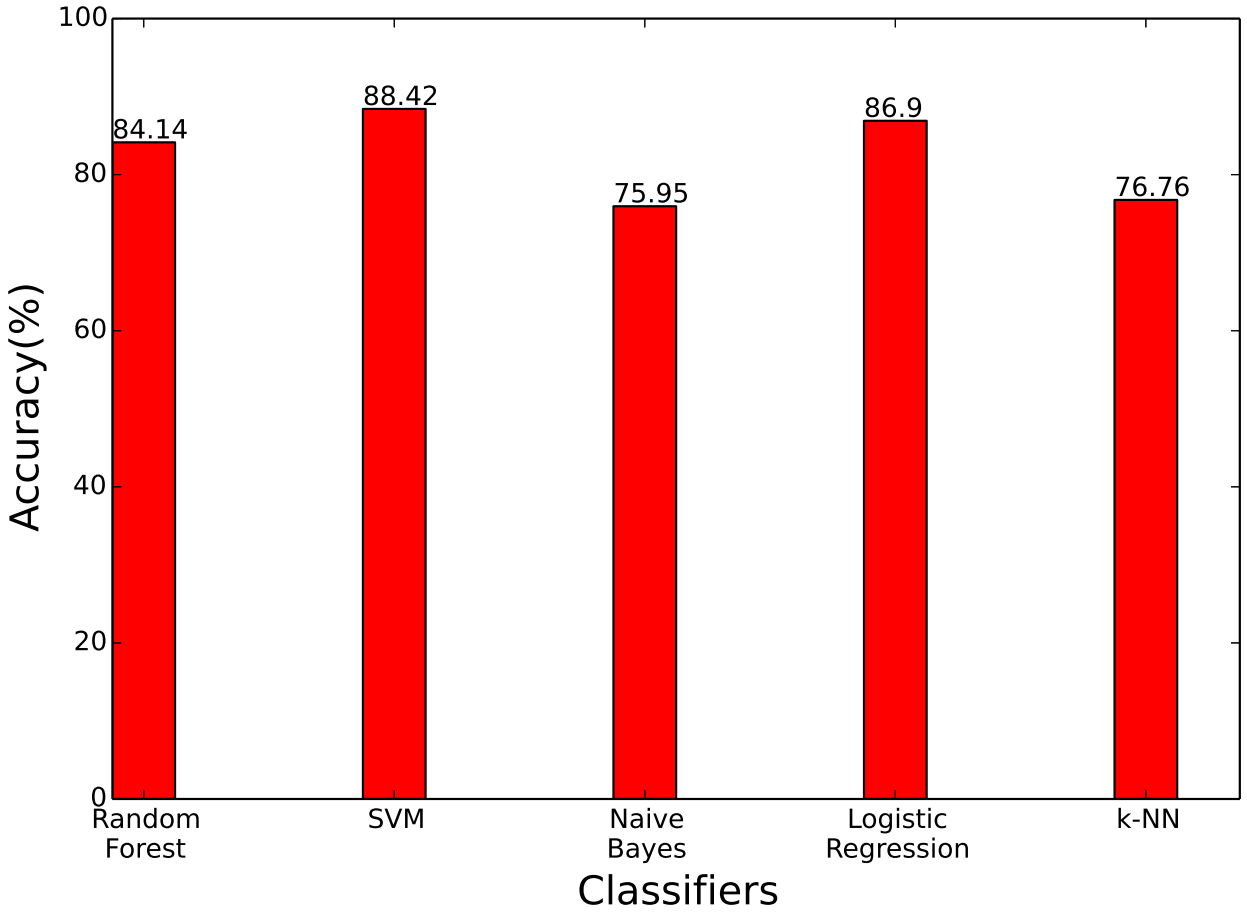


Figure 6 : Accuracies of Different Classifiers with Average Word Vectors on IMDB Dataset.

# Result on English Dataset

| Features | Accuracy |
|---|---|
| Dredze et al.(2008) | 85.90 |
| Max Entropy | 83.79 |
| WordVector Averaging (Our Method) | 89.41 |
| Composite Document Vector(Our Method) | 92.17 |
| Composite Document Vector+RNNLM | **92.91** |

Table 7 : Results on Amazon Electronics Review Dataset

# Result on Hindi Dataset

| Features | Accuracy(1) | Accuracy(2) |
|---|---|---|
| WordVector Averaging | 78.0 | 79.62 |
| WordVector+tf-idf | 90.73 | 89.52 |
| WordVector+tf-idf without stop words | 91.14 | 89.97 |
| Weighted WordVector | 89.71 | 85.90 |
| Weighted WordVector+tf-idf | **92.89** | **90.30** |

Table 8 : Accuracies for Product Review and Movie Review Datasets.

# Result on Hindi Dataset

| Experiment | Features | Accuracy |
|---|---|---|
| Subjective Lexicon (Bakliwal et al.(2012)) | Simple Scoring | 79.03 |
| Hindi-SWN Baseline (Arora et al.(2013)) | Adjective and Adverb presence | 69.30 |
| Word Vector with SVM (Our method) | tf-idf with word vector | 91.14 |
| Weighted Word Vector with SVM (Our method) | tf-idf+weighted word vector | **92.89** |

Table 9 : Comparison of Approaches: Product Review Dataset

| Experiment | Features | Accuracy |
|---|---|---|
| In language using SVM (Joshi et al.(2010)) | tf-idf | 78.14 |
| MT Based using SVM (Joshi et al.(2010)) | tf-idf | 65.96 |
| Improved Hindi-SWN (Bakliwal et al.(2012)) | Adjective and Adverb presence | 79.0 |
| WordVector Averaging | word vector | 78.0 |
| Word Vector with SVM (Our method) | tf-idf; word vector | 89.97 |
| Weighted Word Vector with SVM (Our method) | tf-idf+weighted word vector | **90.30** |

Table 10 : Comparison of Approaches: Movie Review Dataset

# Odd One Out

| | | | |
|---|---|---|---|
| breakfast | **cereal** | lunch | dinner |
| eight | seven | **owe** | nine |
| **shopping** | math | reading | science |

Table 11 : Odd One Out in English

| | | | |
|---|---|---|---|
| भारत | **मुम्बई** | रूस | चीन |
| लड़की | बेहन | **मर्द** | महिला |
| **उद्योग** | नेता | मंत्री | सरकार |

Figure 7 : Odd One Out in Hindi

# Similar Words

| Father | France | XBOX | scratched | megabits |
|---|---|---|---|---|
| grandfather | Germany | XBLA | scraped | gigabits |
| uncle | French | Xbox360 | rubbed | kilobits |
| mother | Greece | SmartGlass | bruised | megabit |
| father-in-law | Netherlands | 360/PS3 | cracked | terabits |
| brother | Scotland | XBLA | discarded | MB/s |
| - | - | Qubed | shoved | Tbit/s |
| - | - | Kinect | tripped | - |

Table 12 : Top Few Similar words in English

| भारत | व्यापार | ओबामा |
|---|---|---|
| प्रदेश | व्यवसाय | क्लिंटन |
| तिब्बत | पुनर्बीमा | बराक |
| देश | वाणिज्य | सीनेटर |
| आँध्रप्रदेश | बैंकिंग | राष्ट्रपति |
| लद्दाख | उद्योग | उम्मीदवार |

Figure 8 : Top Few Similar words in Hindi

# Outline

# Conclusion

1. We present a unsupervised language independent model that
   - overcomes the problems of BOW models
   - gives individual importance to words as well as sentences as a whole
2. We overcome the problems of language dependent models such as Recursive Tensor Neural Network(Socher et al., 2013)
3. We release a larger and more generic dataset of Hindi movie reviews
4. We improve the state-of-the-art results on sentiment analysis
   - On the IMDB dataset, we improve by 1.6%
   - On the Amazon electronics dataset, we improve by 7.01%
   - On the Hindi product and movie reviews, we improve by 13.86% and 11.30% respectively

# Conclusion

1. We present a unsupervised language independent model that
   - overcomes the problems of BOW models
   - gives individual importance to words as well as sentences as a whole
2. We overcome the problems of language dependent models such as Recursive Tensor Neural Network(Socher et al., 2013)
3. We release a larger and more generic dataset of Hindi movie reviews
4. We improve the state-of-the-art results on sentiment analysis
   - On the IMDB dataset, we improve by 1.6%
   - On the Amazon electronics dataset, we improve by 7.01%
   - On the Hindi product and movie reviews, we improve by 13.86% and 11.30% respectively

# Conclusion

1. We present a unsupervised language independent model that
   - overcomes the problems of BOW models
   - gives individual importance to words as well as sentences as a whole
2. We overcome the problems of language dependent models such as Recursive Tensor Neural Network(Socher et al., 2013)
3. We release a larger and more generic dataset of Hindi movie reviews
4. We improve the state-of-the-art results on sentiment analysis
   - On the IMDB dataset, we improve by 1.6%
   - On the Amazon electronics dataset, we improve by 7.01%
   - On the Hindi product and movie reviews, we improve by 13.86% and 11.30% respectively

# Conclusion

1. We present a unsupervised language independent model that
   - overcomes the problems of BOW models
   - gives individual importance to words as well as sentences as a whole
2. We overcome the problems of language dependent models such as Recursive Tensor Neural Network(Socher et al., 2013)
3. We release a larger and more generic dataset of Hindi movie reviews
4. We improve the state-of-the-art results on sentiment analysis
   - On the IMDB dataset, we improve by 1.6%
   - On the Amazon electronics dataset, we improve by 7.01%
   - On the Hindi product and movie reviews, we improve by 13.86% and 11.30% respectively

# Future Work

1. Better Composition Methods
2. Enhanced document vectors has given an indication that current representations are not sufficient to model documents and that ensembles could also prove useful in tasks such as sentiment analysis
3. Region of Importance in NLP where we filter out sentiment oriented sentences and phrases from a unfocused corpus which contains text from various domains

# Future Work

1. Better Composition Methods
2. Enhanced document vectors has given an indication that current representations are not sufficient to model documents and that ensembles could also prove useful in tasks such as sentiment analysis
3. Region of Importance in NLP where we filter out sentiment oriented sentences and phrases from a unfocused corpus which contains text from various domains

# Future Work

1. Better Composition Methods
2. Enhanced document vectors has given an indication that current representations are not sufficient to model documents and that ensembles could also prove useful in tasks such as sentiment analysis
3. Region of Importance in NLP where we filter out sentiment oriented sentences and phrases from a unfocused corpus which contains text from various domains

Thank you!