Word Embeddings

CS 6956: Deep Learning for NLP

THE UNIVERSITY OF UTAH

Overview

- Representing meaning
- Word embeddings: Early work
- Word embeddings via language models
- Word2vec and Glove
- Evaluating embeddings
- Design choices and open questions

Overview

- Representing meaning
- Word embeddings: Early work
- Word embeddings via language models
- Word2vec and Glove
- Evaluating embeddings
- Design choices and open questions

What we have seen

- Basic algorithms (i.e. circa 2014) for word embeddings
- Some hints about evaluation

Several design choices/extensions

1. What is a good context?

2. Can we use syntactic windows?

3. How to pre-process the text before training?

4. Multilingual embeddings

5. Character-based/subword embeddings

1. Impact of contexts

- Context window size: Should we use large or small context windows?
  - Large context windows makes topically similar words closer (eg: sport, baseball, referee, etc are grouped)
  - Smaller context windows focus on syntactic or functional similarities (eg: batting, running, jumping, etc are grouped)

1. Impact of contexts

- Context window size: Should we use large or small context windows?
  - Large context windows makes topically similar words closer (eg: sport, baseball, referee, etc are grouped)
  - Smaller context windows focus on syntactic or functional similarities (eg: batting, running, jumping, etc are grouped)

- Positional contexts: Should the context features be different for words in different positions?
  - Eg: For word 1, if the previous word is `cat`, and for word 2, `cat` appears two words after it, should both instances of cat be treated similarly?
    - Or should they be treated differently by encoding the position in the context?
  - Positional contexts seem to help if we care about grouping syntactic function or words with similar parts-of-speech

2. Syntactic windows

Idea: Instead of using proximal words in the sentence, use the dependency tree to decide on which words are proximal [Bansal et al 2014, Levy and Goldberg 2014, etc]

John saw a great white shark with his telescope

2. Syntactic windows

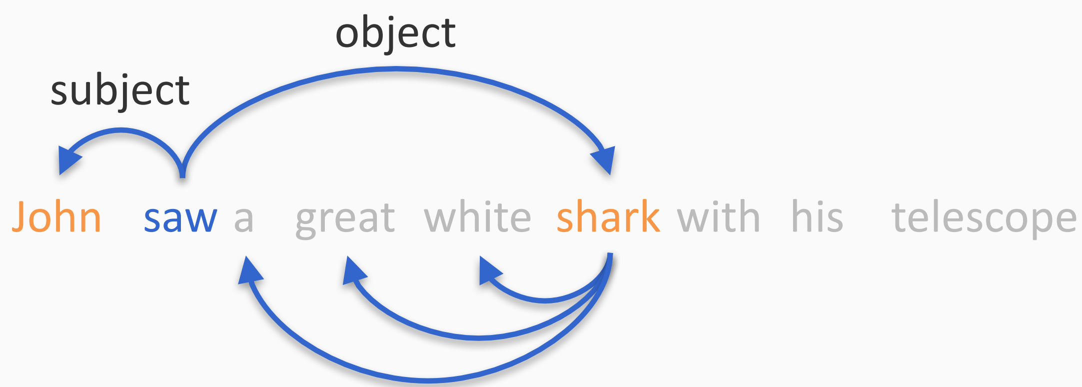
Idea: Instead of using proximal words in the sentence, use the dependency tree to decide on which words are proximal [Bansal et al 2014, Levy and Goldberg 2014, etc]

John saw a great white shark with his telescope

Word context may be less relevant

2. Syntactic windows

Idea: Instead of using proximal words in the sentence, use the dependency tree to decide on which words are proximal [Bansal et al 2014, Levy and Goldberg 2014, etc]



Dependency context may offer more information

3. Preprocessing text for word embeddings

- Several choices available
  - Should the words be lemmatized?
    - `good`, `better`, `best` map to *`good`*
    - `give`, `gives`, `giving`, `gave`, etc map to *`give`*
  - Should words retain their capitalization?
    - Eg: Should `Apple` and `apple` be treated as the same word?
  - Should very rare or frequent words be filtered out?
    - Eg: `of` vs. `octothorpe`
  - Should some sentences be filtered out?
    - Eg: Long sentences, short sentences
- And many more. Can be treated as hyperparameters

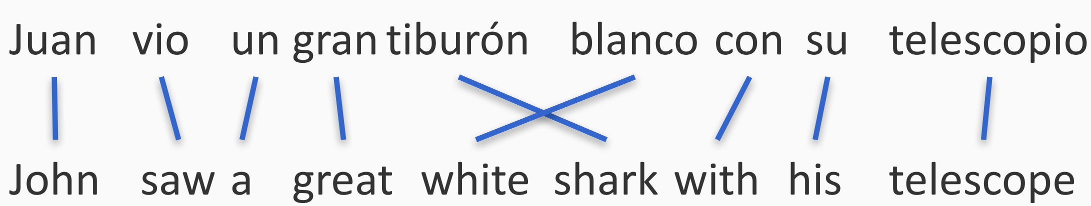
4. Multilingual embeddings

Define context using translations

- [Faruqui and Dyer, 2014, Hermann and Blunsom, 2014]

General idea:

- Align two languages using an off-the-shelf bilingual aligner
  - Eg: from Giza++
- Context = words in the other language it aligns with



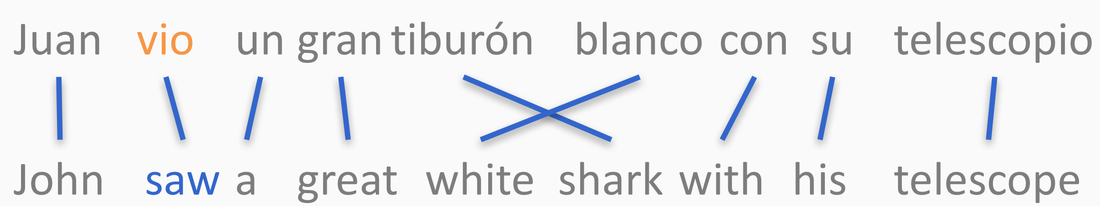
4. Multilingual embeddings

Define context using translations

– [Faruqui and Dyer, 2014, Hermann and Blunsom, 2014]

General idea:

– Align two languages using an off-the-shelf bilingual aligner
  - Eg: from Giza++
– Context = words in the other language it aligns with



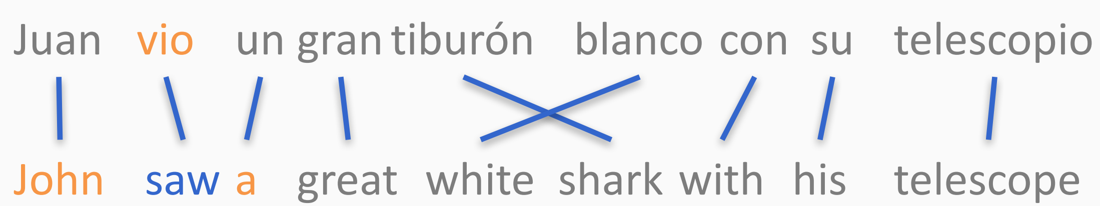
4. Multilingual embeddings

Define context using translations

– [Faruqui and Dyer, 2014, Hermann and Blunsom, 2014]

General idea:

– Align two languages using an off-the-shelf bilingual aligner
  - Eg: from Giza++
– Context = words in the other language it aligns with + context in a window



4. Multilingual embeddings

Define context using translations

- [Faruqui and Dyer, 2014, Hermann and Blunsom, 2014]

General idea:

- Align two languages using an off-the-shelf bilingual aligner
  - Eg: from Giza++
- Context = words in the other language it aligns with + context in a window
  - Or other variants possible

Somewhat decreases the problem of antonyms getting similar vectors

5. Character-based/subword embeddings

Intuition: The meaning of a word depends its context, but its own sub-units

Example: substitute, substitution, substitutable

Common approaches to capture this rely on featurizing words instead of using one-hot embeddings

- Hand crafted features
- All subwords in the word
- Convolutional or recurrent networks to construct word embeddings

Perhaps especially helpful for

1. Previously unseen words
2. Morphologically rich languages

Problems/ open research questions

- Antonyms tend to be embedded together
- Unclear how similarity is defined
  - `cat` closer to `dog` or `tiger`?
- Embeddings may exhibit gender, racial, ethnic and other social biases
  - Eg: female names are embedded closer to stereotypically female social roles
- Word sense is ignored
  - `bank` could be a financial institution or a river bank
- Obvious things are not talked about in text
  - Eg: most sheep are white, but "`black sheep`" may be more frequent than "`whitesheep`"