Introduction to NLP and Word Embedding

Text Analytics (MLDS 414)

Yuri Balasanov

Northwestern University

Outline of the Session

- NLP in the 20-th century
- Distributional Semantics
- Word senses
- Semantic relationships: synonymity, antonymity, and taxonomic relationships
- WordNet
- Word sense disambiguation
- Neural networks and early embeddings: Word2Vec, FastText, GloVe
- Risk of social bias in pretrained embeddings
- From fixed embedding to context
    - Transfer learning from pretrained models

Main texts:

Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Third Edition draft. Daniel Jurafsky, James H. Martin, © 2023

NLP in the 20th Century I

- Use of statistics in NLP started in 1980s. That was the time when subjects like Mathematical Linguistics, Computational linguistics were born

- Initially the data were collected manually by interviewing **informants**, i.e. native language speakers providing linguistic data and insights.
- The data were written on IBM punch cards stacked in shoe boxes

NLP in the 20th Century II

- The analysis was based on simple features extracted using frequencies of word or n-grams in the observed documents, and measures like TF-IDF
- These features were then used with standard Machine Learning methods like logistic regression, naive Bayes, k-nearest neighbors, hidden Markov models, random forests, support vector machines
- To capture deeper structure of the language there were products like WordNet
- WordNet is a lexical database of semantic relations (synonyms, hyponyms, meronyms) between words. It started in 1985 at Cognitive Science Laboratory at Princeton University. It was first created in English only, but today it covers more than 200 languages.

Distributional Semantics

- Word **embeddings** into vectors appeared in 1990s: first in the form of high-dimensional and sparse one-hot encoding vectors, and then continuous representations
- Deep theory providing foundation for word embeddings is called **Distributional Semantics**
- **Distributional Semantics** is a computationally implementable theory of meaning that started developing in 1950-s
- The basic idea of Distributional Semantics is called **Distributional Hypothesis**:
  - **Words that are used and occur in the same contexts tend to have similar meanings**
- Distributional similarity between words is mapped into similarity of high dimensional vectors which are continuous vector representations

Distributional Semantics Models

- Initial models for estimating continuous word representations based on Distributional Semantics were statistical
- Latent Semantic Analysis (LSA) analyzes co-occurencies matrices of word counts per document

$$\begin{array}{cccc} & \text{Doc}_1 & \cdots & \text{Doc}_M \\ \text{Word}_1 & X_{1,1} & \cdots & X_{1,M} \\ \vdots & \vdots & \ddots & \vdots \\ \text{Word}_N & X_{N,1} & \cdots & X_{N,M} \end{array}$$

- Reduction of dimensionality with method of Singular Value Decomposition (SVD) allows extracting **term-to-term**, **document-to-document**, and **term-to-document** similarities
- Drawbacks:
  - Every change of the dictionary requires recalculation of the model
  - Sensitivity to word frequency imbalance forces cumbersome preprocessing, like removal of stop words and normalization
  - Computationally expensive, as dictionary and/or corpus grow large

Word Senses

- Words are ambiguous: some of them may mean different things, i.e. they are **polysemous**

Definition

A discrete representation of one aspect of the meaning of a word is called its **sense**

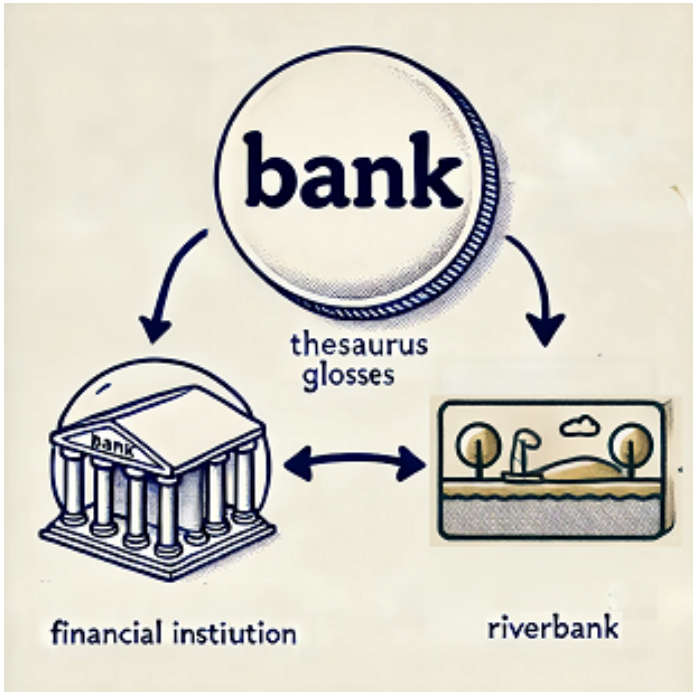
- The task of determining which sense of a word is being used in a particular context is called **word sense disambiguation** (**WSD**)
- WSD is very important for many NLP tasks, helping to understand the context or translating into other language

Example

In Spanish the animal "bat" is a "murci´elago" while the baseball "bat" is a "bate"

WSD with Thesaurus

- One way of finding the word sense in a particular context is based on looking for textual definitions called **gloss** in the dictionary or thesaurus



Example

The two senses of "bank" are described in thesaurus:
1. Financial institution that accepts deposits and channels the money into lending activities
2. Sloping land (especially the slope beside a body of water)

- Glosses are **not formal definitions** but they are useful for NLP algorithms because they can be used for calculation embeddings, telling something about the meaning of the sentence

Synonyms I

- When two words have same senses they a called **synonyms**
- Note that synonymity is a **relationship between senses but not the words**, some senses of words may be synonymous while others are not

Example

Words "big" and "large" may seem synonymous. In the following 2 sentences they can replace each other preserving the truth of the statement

- How big is that plane?
- Would I be flying on a large or small plane?

Synonyms II

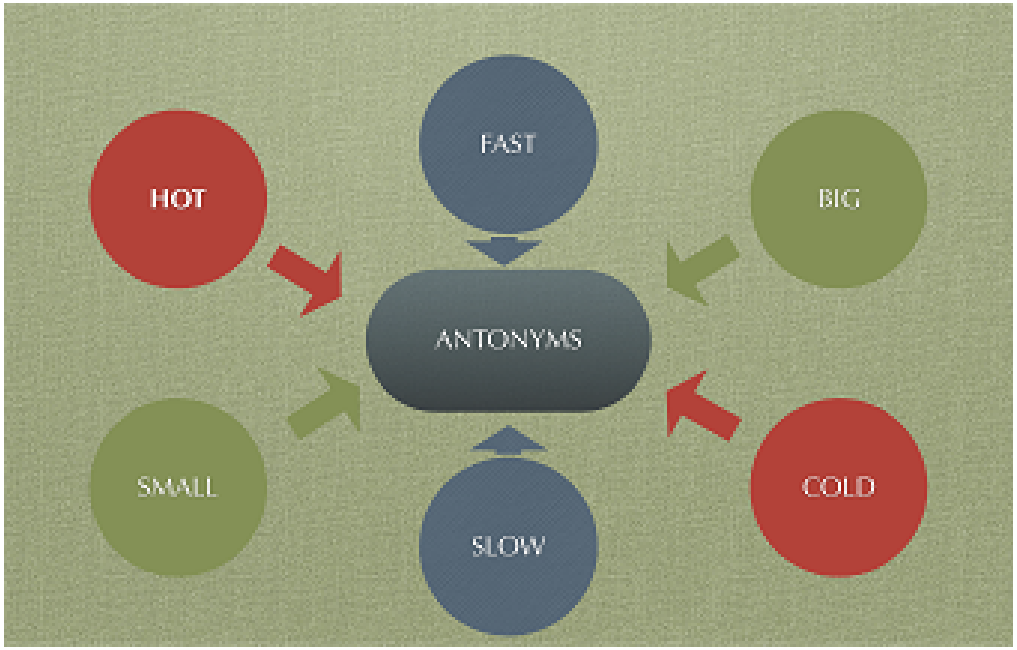
Example

But in the other 2 sentences the words are not substitutable

- Miss Nelson, for instance, became a kind of big sister to Benjamin
- Miss Nelson, for instance, became a kind of large sister to Benjamin

The word "big" has a sense of being older or grown up, while "large" lacks this sense

Antonyms

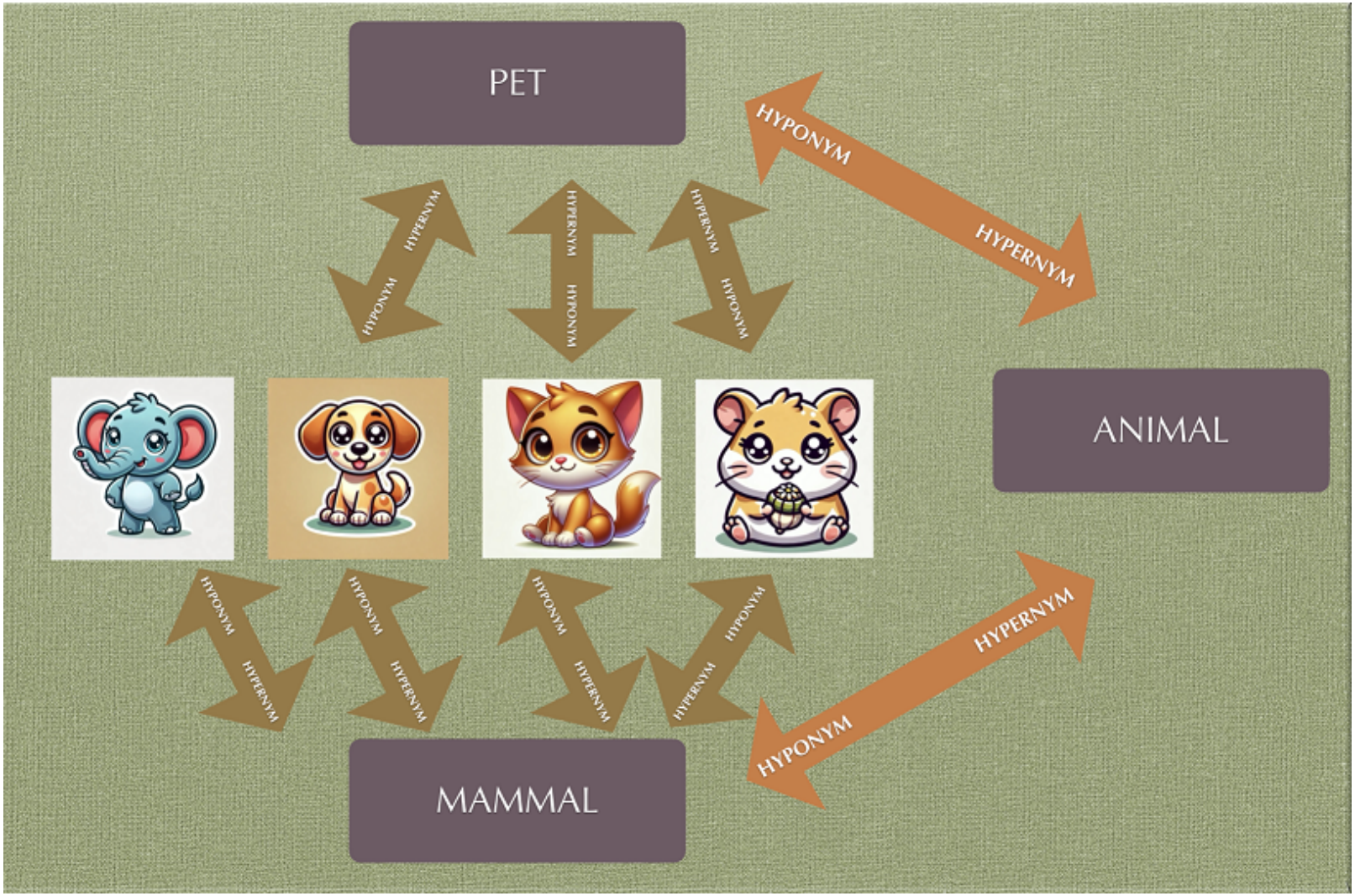


- **Antonyms** are words with an opposite meaning, like "long"/"short", "big"/"little", "fast"/"slow", "rise"/"fal", "up"/"down", "in"/"out"
- Two senses can be antonyms if
  - They define a binary opposition of some scale ("long"/"short", "big"/"little", "fast"/"slow")
  - They are **reversives**: describe change or movement in opposite directions ("rise"/"fal", "up"/"down", "in"/"out")

Taxonomic Relationships I

- **Word senses can be related taxonomically**
- One word (sense) is a **hyponym or subordinate** of another if it is more specific, denoting a subclass of the other
  - "car" = **hyponym**("vehicle"), "dog" = **hyponym**("animal")
- The opposite taxonomic relationship is **hypernym** or **superordinate**:
  - "vehicle" = **hypernym**("car"), "animal" = **hypernym**("dog")
- **Hypernymy through entailment** can also be defined as: $\forall x$ $A(x) \Rightarrow B(x)$. Example: "eating" entails "chewing"

Taxonomic Relationships II



WordNet

- The most common resource for sense relations in English and many other languages is the **WordNet** lexical database (Fellbaum, 1998)
- English WordNet contains three separate databases, for nouns, for verbs, and for adjectives and adverbs
- Each database contains lemmas, each lemma has a set of senses
- For each lemma, each sense has a gloss, a list of synonyms, and sometimes also usage examples
- The set of near-synonyms for a WordNet sense is called a **synset**

Example

Some of the senses for noun "bass" in WordNet:

1. bass^1 - (the lowest part of the musical range)
2. bass^3, basso^1 - (an adult male singer with the lowest voice)
3. sea bass^1, bass^4 - (the lean flesh of a saltwater fish of the family Serranidae)

Polysemous Lemma "bank"

Example

- Sense 1:
  - **Gloss**: A financial institution that accepts deposits and channels the money into lending activities.
  - **Usage Example**: "He cashed a check at the bank."
- Sense 2:
  - **Gloss**: The land alongside or sloping down to a river or lake.
  - **Usage Example**: "The river overflowed its bank."
- Sense 3:
  - **Gloss**: A long pile or heap; mass.
  - **Usage Example**: "A bank of snow."
- Sense 4:
  - **Gloss**: A slope in a road or track; used especially to provide a banking effect in turns.
  - **Usage Example**: "The car raced around the banked curve.

Sense Relations in WordNet

- WordNet represents different types of relationships between senses
- The relationships are defined by the graph with vertical and horizontal connections

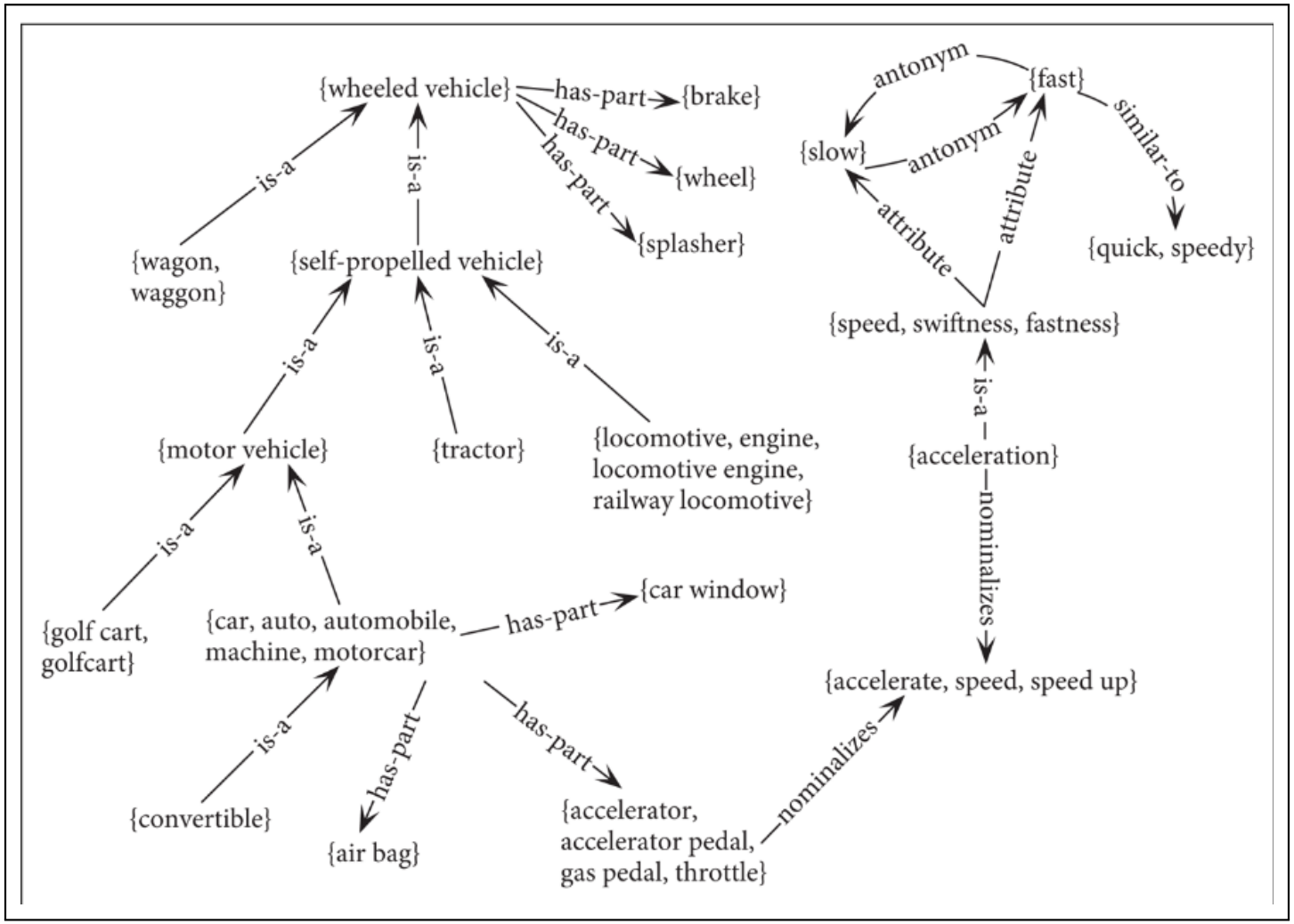
Example

WordNet represents hyponymy by relating each synset to its immediately more general and more specific synsets:

- "self-propelled vehicle" is a hyponym to "wheeled vehicle"
- "self-propelled vehicle" is a hypernym to "tractor"

Longer chains can also be formed in both directions: "convertible"⇒["car", "auto", "automobile", "machine", "motorcar"]⇒"motor vehicle"⇒"self-propelled vehicle"⇒"wheeled vehicle"

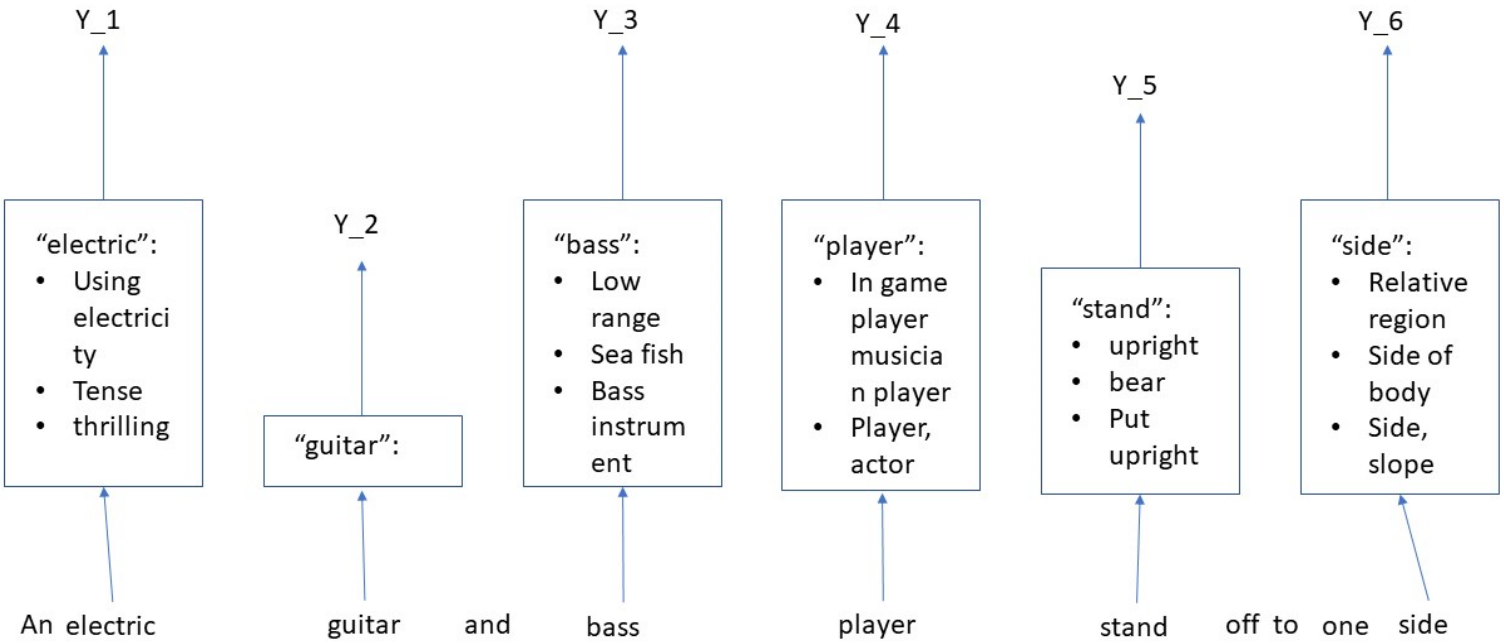
Example of WordNet Graph



WSD with WordNet I

- WordNet is useful for WSD.
- The problem of WSD is solved by selecting the most appropriate sense out of synsets of WordNet.
- A surprisingly strong baseline is simply to choose the most frequent sense for each word from the senses of a labeled corpus.
- For WordNet, this corresponds to the first sense, since senses in WordNet are ordered from most frequent to least frequent.
- The baseline algorithm gives more than 90% accuracy. The rest of improvement is achieved by other, more sopisticated methods.
- One of the best performing WSD algorithms is a simple 1-nearest-neighbor algorithm using contextual word embeddings.

WSD with WordNet II



One Sense per Discourse

- To increase efficiency of the WSD process it is common to use an heuristic method, called **one sense per discourse (OSpD),** based on the observation that a word appearing multiple times in a text or discourse often appears with the same sense.

Example

John decided it was time to open a savings account at the **bank**. He had been working hard over the past few years and wanted a secure place to store his money. When he arrived at the **bank**, the teller greeted him with a warm smile and asked how she could assist him. John explained that he wanted to learn more about the interest rates offered by the **bank**. The **bank** manager was called over to discuss the various savings options available. After a detailed conversation, John was satisfied and decided to deposit his money into the **bank**.

- **OSpD** principle: once the sense of "bank" is established in the discourse, it is assumed to carry the same sense throughout the text.

Neural Networks and Early Embeddings

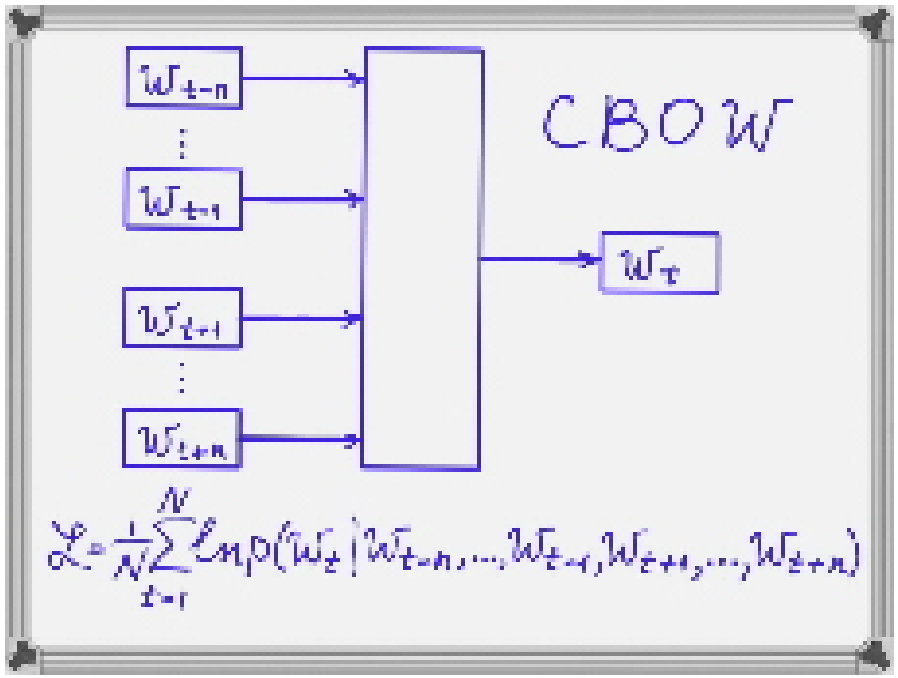
- First applications of neural networks to NLP started in 2000s
- By 2010 the power of pretrained word embeddings was clearly demonstrated: Colbert&Weston (2008) showed word embedding trained on large enough data capture syntactic, morphological, contextual or semantic similarities
- In 2013 T. Mikolov et al. from Google created Word2Vec, a neural network model for pre-training word embeddings
- A year later Pennington et al. from Stanford University created Global Vectors (GloVe) for word representations
- Dense vector representations obtained using unsupervised neural network models became state-of-the-art approach
- Embeddings are obtained as weights of the first embedding layer of neural network

Word2Vec

- Word2Vec appeared in 2013 at Google (T. Mikolov, K. Chen,G. Corrado, J. Dean). It was much less computationally expensive than its predecessors
- Word2Vec contains 2 model architectures representing different approaches to learning: Continuous Bag-Of-Words (CBOW) and Continuous Skip-Gram
- Unlike previous models both architectures do not have costly hidden layers
- Word2Vec models also take into account the context of the word
- Word2Vec requires a lot of data to train, for example the entire Wikipedia corpus
- Word2Vec is window-based, it does not use the information from the context larger than the window
- Word2Vec cannot embed Out-Of-Vocabulary (OOV) words

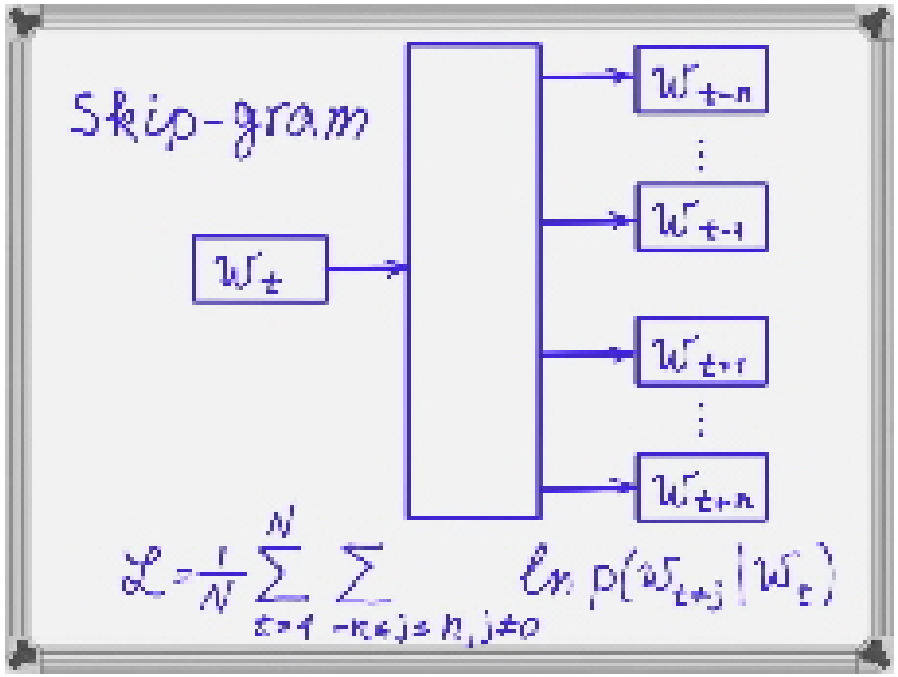
Word2Vec: CBOW

- The CBOW model learns to predict the target word from the n words before and n words after it
- Continuous Bag-of-Words makes a continuous representation of the word from the bag-of-words in which the order is not important
- The model does not have to be as accurate as any language model because it only makes words representations



Word2Vec: Skip-gram

- The Skip-gram model predicts the words surrounding the target word from that word
- Like CBOW the Skip-gram model uses only one hidden layer and cannot capture the depth of language
- However, it provides embeddings capturing some semantic similarity



FastText

- FastText was created at Facebook in 2016 (Bojanowski et al.). It builds up on Word2Vec while improving its efficiency
- FastText uses the Skip-gram model from Word2Vec
- A word in FastText is associated with all n-grams containing that word. Then vector representation of the word is formed as sum of n-gram representations
- Because it is based on *n*-grams it can embed OOV by calculating the score of an unknown word from all n-grams containing it
- FastText still does not deal with the disambiguation issue: each word embedding is not created from the context and can have only one meaning

GloVe

- Global Vectors for Word Representation (GloVe) was created in 2014 at Stanford University by Pennington et al
- GloVe is trained as a language model by predicting words
- GloVe is based on ratios of co-occurrence probabilities calculated over the whole corpus of training texts. These ratios are mapped into differences between the representation vectors

Risk of Social Bias in Pretrained Embeddings I

- Pretrained embeddings models require very large amount of data for training. Typically, for that class of models the training set is the entire Wikipedia
- But such large decentralized and not strictly edited corpus almost certainly reflects some social biases. These biases will be learned by the model and reproduced in any inference made with the models

Examples

1. It was discovered in 2016 that Word2Vec embeddings resulted in the following proportion revealing gender stereotype: "man" relationship to "computer scientist" has similarity to relationship between "woman" and "homemaker"
2. Another stereotype shown in the workshop is that relationship between "father" and "doctor" has almost 70% similarity with "mother" - "nurse"

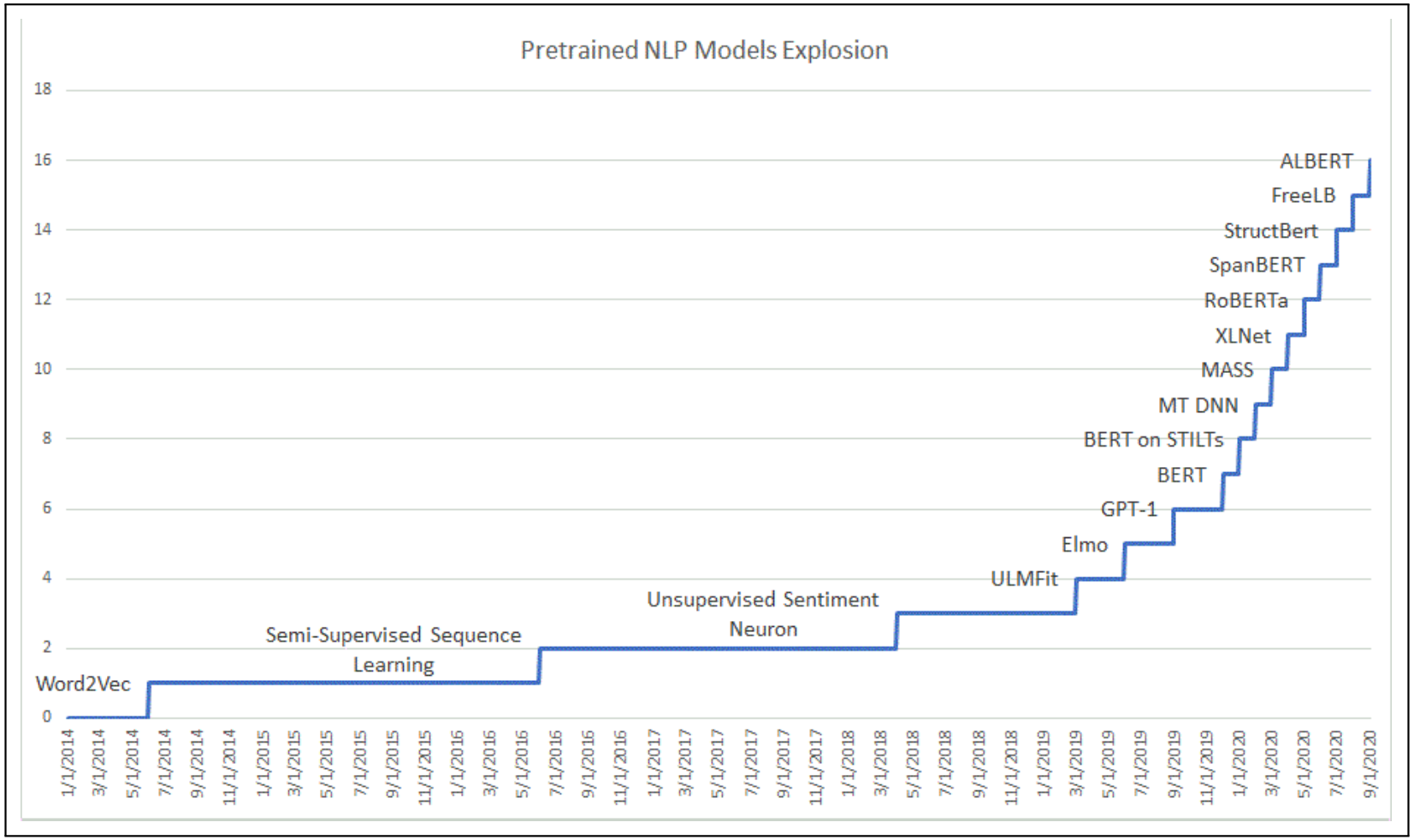
Risk of Social Bias in Pretrained Embeddings II

- Such biases can lead to so-called **allocational harm**, when a system allocates resources (jobs or credit) unfairly to different groups
- For example algorithms using NLP with embeddings as part of a search for hiring potential programmers or doctors might incorrectly downweight documents with women's names
- Embeddings may not just reflect the bias in the data, but amplify it
- Embeddings also encode the implicit associations, consciously or unconsciously used by humans.
- Cosine measure between embeddings, for example associate African-American names with unpleasant words, unlike European-American names, associate male names more with mathematics and female names with the arts, or associate old people's names with unpleasant words
- implicit associations encoded in embeddings are an example of a **representational harm** - a harm caused by a system demeaning or even ignoring some social groups

From Fixed Embedding to Context

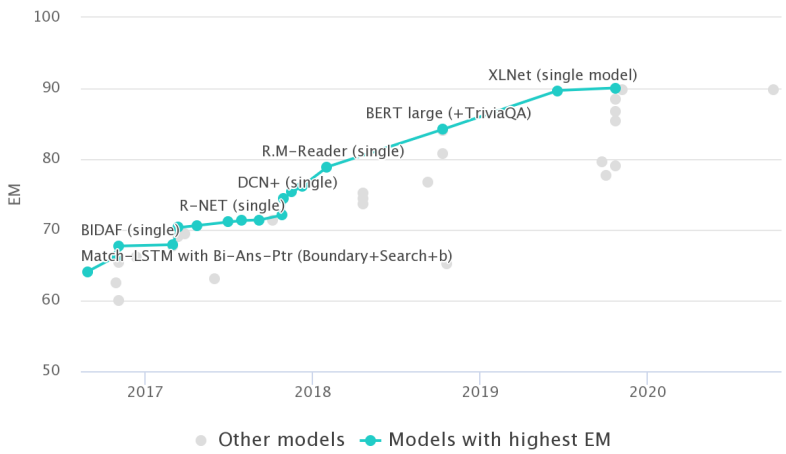
- Models for word embedding, like Word2Vec, GloVe or FastText help understanding language much better than one-hot embedding
- These models produce dense embeddings with much lower dimensionality
- They are more efficient than the first neural network-based embedding methods of 2000-s
- They also can be pre-trained on large amounts of data
- However, they have one significant disadvantage: each word is mapped to only one vector and thus can have only one meaning
- New generation of embedding models that appeared since 2018 assign embedding vectors based on the context

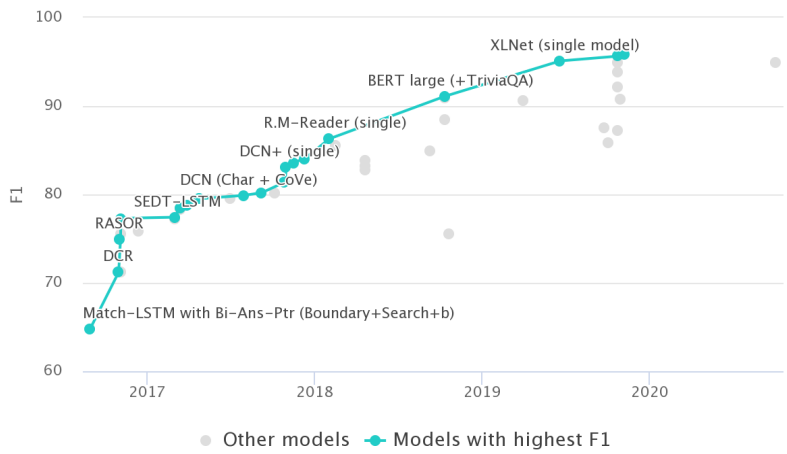
Gradually, Then Suddenly



Transfer Learning in NLP has been brewing for some time. Then it exploded. The explosion graph is inspired by one of the lectures by Collin Raffel

Above Human Ability



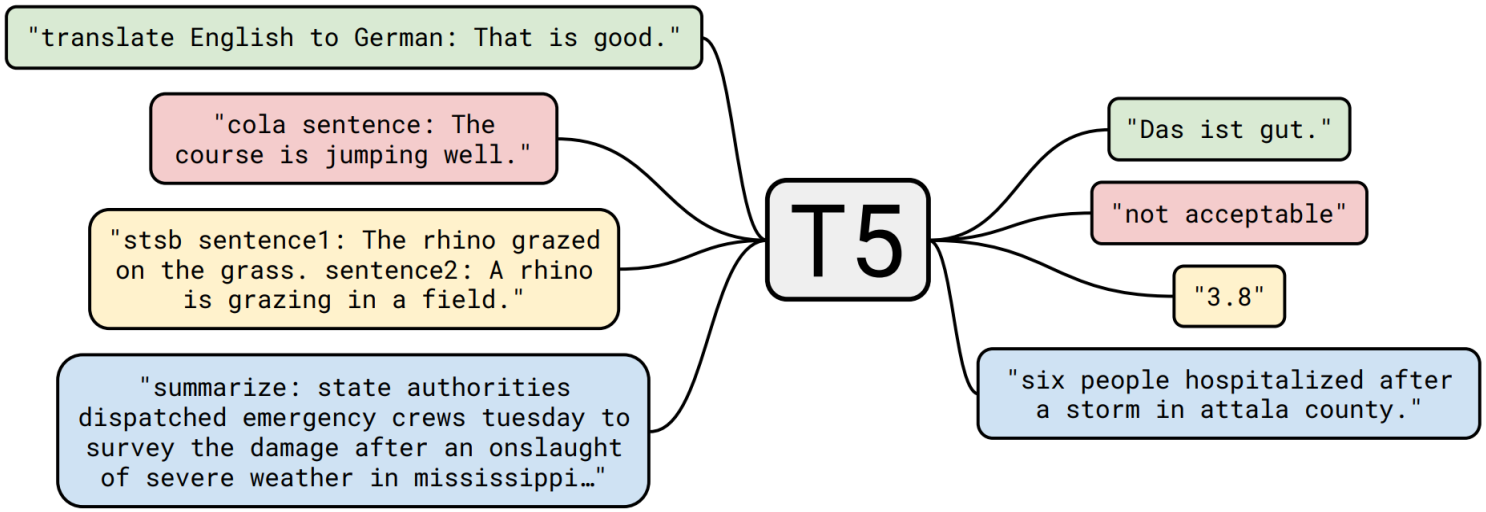


The latest models are claimed to exceed human level of performing similar downstream tasks

Source:

https://paperswithcode.com/sota/question-answering-on-squad11-dev

Transfer Learning



Source: https://1.bp.blogspot.com/-89OY3FjN0N0/XIQI4PEYGsI/AAAAAAAAFW4/knj8HFuo48cUFlwCHuU5feQ7yxfsewcAwCLcBGAsYHQ/s1600/image2.png