

Information Retrieval (IN4325)

Intro to Word Embeddings

Admin

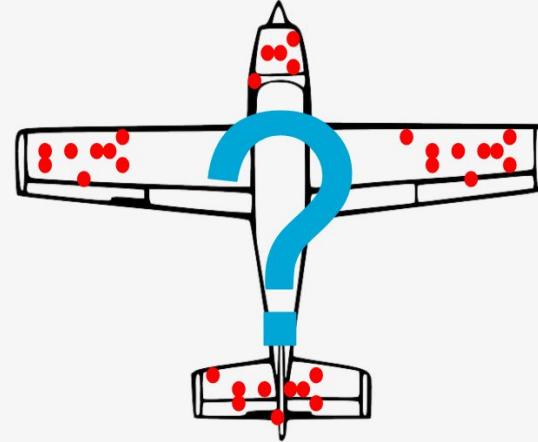
Project

- Plagiarism check, Turnitin
- Replication results, then add features
- Make the pipeline explicit
- Motivate your choices

Yesterday

NLP bias

- Selection bias
- Annotation bias
 - Inter-annotator agreement
 - Bias in feature learning
- Model bias
- System/evaluation bias



		Coder B	
		Treats	No Treats
Coder A	Treats	12	5
	No Treats	8	20

The Modelers' Hippocratic Oath

- ~ I will remember that I didn't make the world, and it doesn't satisfy my equations.
- ~ Though I will use models boldly to estimate value, I will not be overly impressed by mathematics.
- ~ I will never sacrifice reality for elegance without explaining why I have done so.
- ~ Nor will I give the people who use my model false comfort about its accuracy. Instead, I will make explicit its assumptions and oversights.
- ~ I understand that my work may have enormous effects on society and the economy, many of them beyond my comprehension.

This week

- Why word embeddings
- Word Embeddings
 - Dimensionality reduction, e.g., SVD
 - Word2Vec
 - Skipgrams
 - Continuous Bag of Words (CBoW)
 - GLOVE
- Example: Analogies
 - Limitations
 - Data limitations

To compare pieces of text

- We need effective representation of :
 - Words
 - Sentences
 - Text
- **Approach 1:** Use existing thesauri or ontologies like WordNet and Snomed CT (for medical). Drawbacks:
 - Manual
 - Not context specific
- **Approach 2:** Use co-occurrences for word similarity. Drawbacks:
 - Quadratic space needed
 - Relative position and order of words not considered

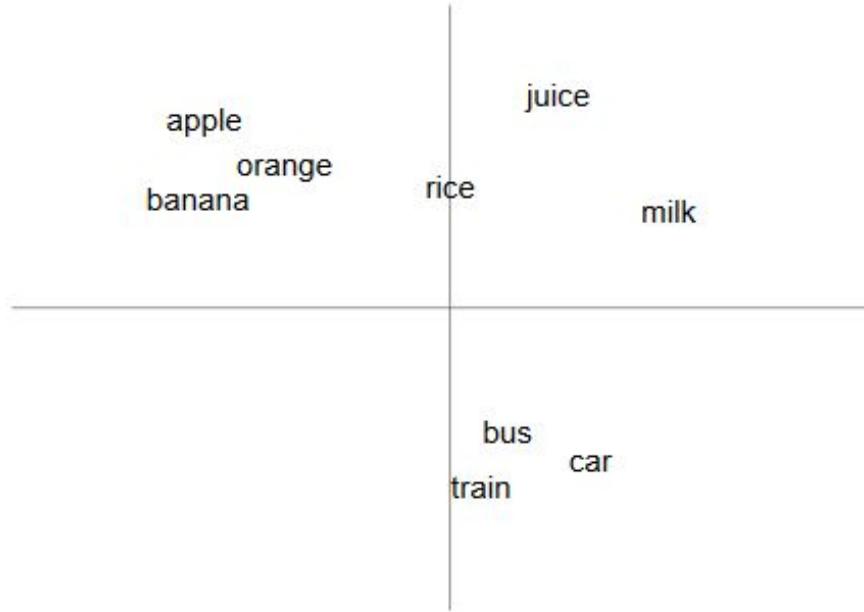
Approach 3: low dimensional vectors

- Store only “important” information in fixed, low dimensional vector.

Word embeddings

“A word is known by the company it keeps”

Distributional semantics



Reference Materials

- Deep Learning for NLP by Richard Socher
(<http://cs224d.stanford.edu/>)
- Word2vec in Gensim by Radim Řehůřek
(<http://rare-technologies.com/deep-learning-with-word2vec-and-gensim/>)

Word Representations

Traditional Method - Bag of Words Model

- Uses one hot encoding
- Each word in the vocabulary is represented by one bit position in a HUGE vector.
- For example, if we have a vocabulary of 10000 words, and “Hello” is the 4th word in the dictionary, it would be represented by: 0 0 0 **1** 0 0 0 0 0
- Context information is not utilized

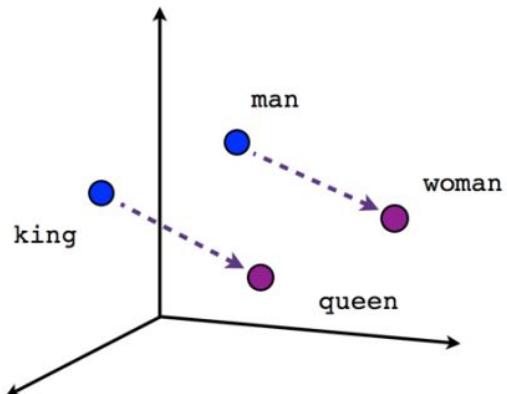
Word Embeddings

- Stores each word in as a point in space, where it is represented by a vector of fixed number of dimensions (generally 300)
- Unsupervised, built just by reading huge corpus
- For example, “Hello” might be represented as :
[0.4, -0.11, 0.55, 0.3 . . . 0.1, 0.02]
- Dimensions are basically projections along different axes, more of a mathematical concept.

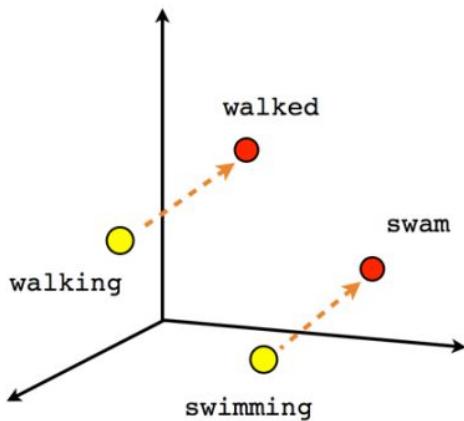
Power of Word Vectors

- They provide a fresh perspective to **MANY** problems in NLP, and not just solve one problem.
- Technological Improvement
 - Rise of deep learning since 2006 (Big Data + GPUs + Work done by Andrew Ng, Yoshua Bengio, Yann Lecun and Geoff Hinton)
 - Application of Deep Learning to NLP – led by Yoshua Bengio, Christopher Manning, Richard Socher, Tomas Mikalov
- The need for unsupervised learning . (Supervised learning tends to be excessively dependant on hand-labelled data and often does not scale)

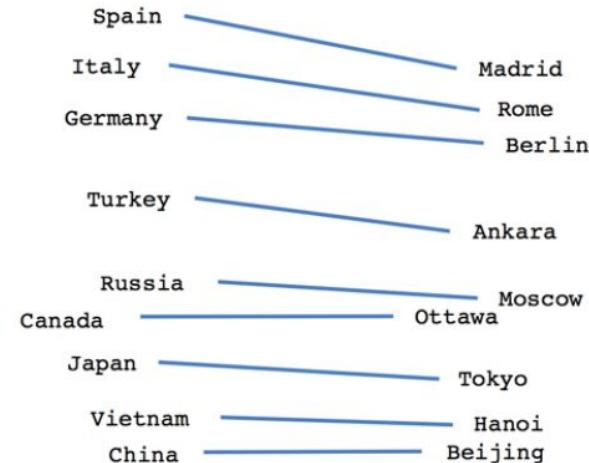
Examples



Male-Female



Verb tense



Country-Capital

$\text{vector[Queen]} = \text{vector[King]} - \text{vector[Man]} + \text{vector[Woman]}$

Examples

<i>Expression</i>	<i>Nearest token</i>
Paris - France + Italy	Rome
bigger - big + cold	colder
sushi - Japan + Germany	bratwurst
Cu - copper + gold	Au
Windows - Microsoft + Google	Android
Montreal Canadiens - Montreal + Toronto	Toronto Maple Leafs

So, how does Word Embedding
‘solve problems in NLP’?

Applications of Word Vectors

Word Similarity

Classic Methods : Edit Distance,
WordNet, Porter's Stemmer,
Lemmatization using dictionaries

- Easily identifies similar words and synonyms since they occur in similar contexts
- Stemming (*thought* -> *think*)
- Inflections, Tense forms
- *E.g., Think, thought, ponder, pondering,*
- *E.g., Plane, Aircraft, Flight*

Applications of Word Vectors

Sentiment Analysis

Classic Methods : Naive Bayes, Random Forests/SVM

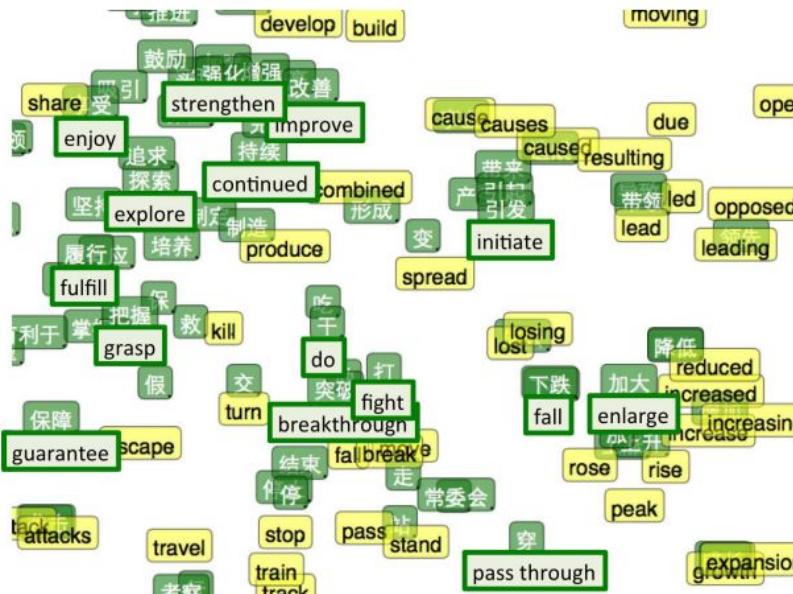
- Classifying sentences as positive and negative
- Building sentiment lexicons using seed sentiment sets
- No need for classifiers, we can just use cosine distances to compare unseen reviews to known reviews.

```
Enter word or sentence (EXIT to break): sad
Word: sad  Position in vocabulary: 4067
-----
          Word      Cosine distance
-----  
saddening           0.727309
      Sad             0.661083
    saddened           0.660439
heartbreaking        0.657351
disheartening       0.650732
  Meny_Friedman     0.648706
parishioner_Pat_Patello 0.647586
    saddens me       0.640712
```

Applications of Word Vectors

Machine Translation

Classic Methods : Rule-based machine translation, morphological transformation



Applications of Word Vectors

Part-of-Speech and Named Entity Recognition

Classic Methods : Sequential Models (MEMM, Conditional Random Fields), Logistic Regression

	POS WSJ (acc.)	NER CoNLL (F1)
State-of-the-art*	97.24	89.31
Supervised NN	96.37	81.47
Unsupervised pre-training followed by supervised NN**	97.20	88.87
+ hand-crafted features***	97.29	89.59

Applications of Word Vectors

Relation Extraction

Classic Methods : OpenIE, Linear programming models,
Bootstrapping

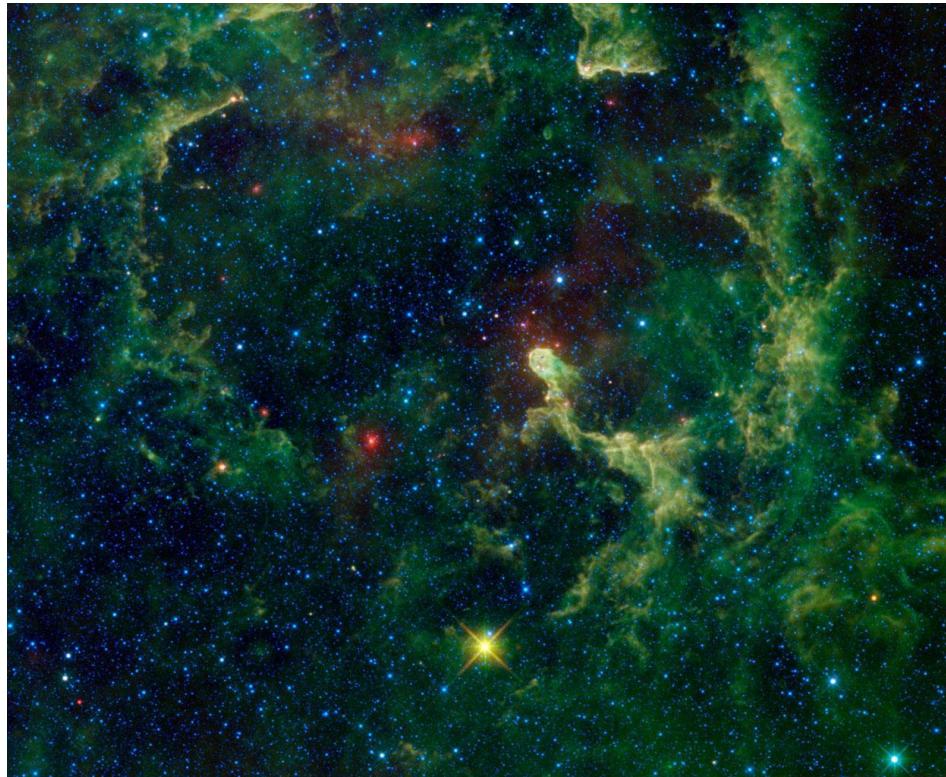
Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

Building these ‘magical’ vectors . . .

- How do we actually build these super-intelligent vectors, that seem to have such magical powers?
- How to find a word’s friends?
- We will discuss the most common methods to build such lower-dimension vector representations for words based on their context
 1. Co-occurrence Matrix with SVD
 2. word2vec (*Google*)
 3. Global Vector Representations (GloVe) (*Stanford*)

Singular Value Decomposition (SVD)

Elephant's
Trunk nebula
from NASA's
Wide-field
Survey
Explorer
(WISE)



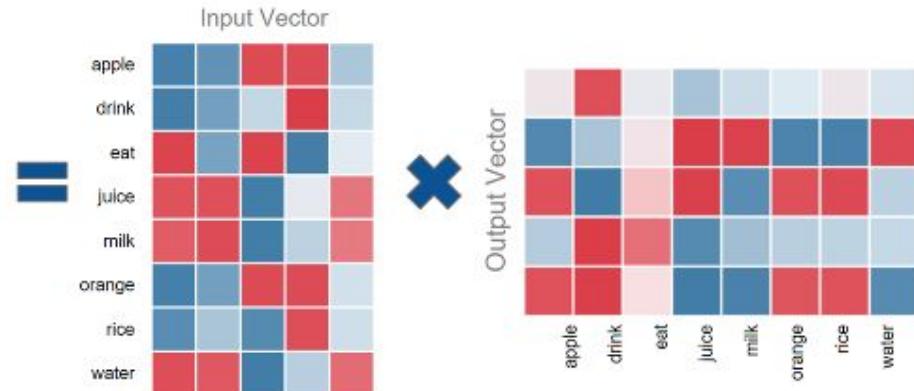
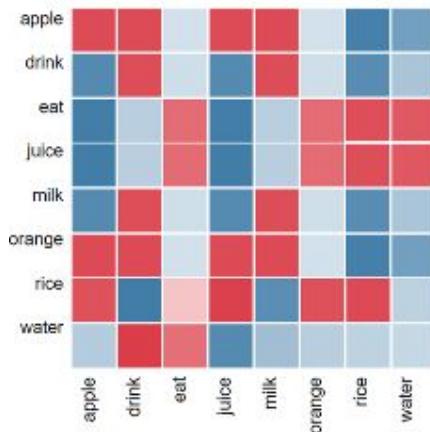
Approach 3: low dimensional vectors

- Store only “important” information in fixed, low dimensional vector.
- Single Value Decomposition (SVD) on co-occurrence matrix
- $k = n = \text{size of vocabulary}$

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

$\begin{matrix} n \times k \\ \end{matrix} = \begin{matrix} n \times k \\ \text{Orthogonal matrix} \end{matrix} \begin{matrix} k \times k \\ \text{Diagonal matrix} \end{matrix} \begin{matrix} k \times k \\ \text{Orthogonal matrix} \end{matrix}$

Singular Value Decomposition



The problem with this method, is that we may end up with matrices having billions of rows and columns, which makes SVD computationally restrictive.

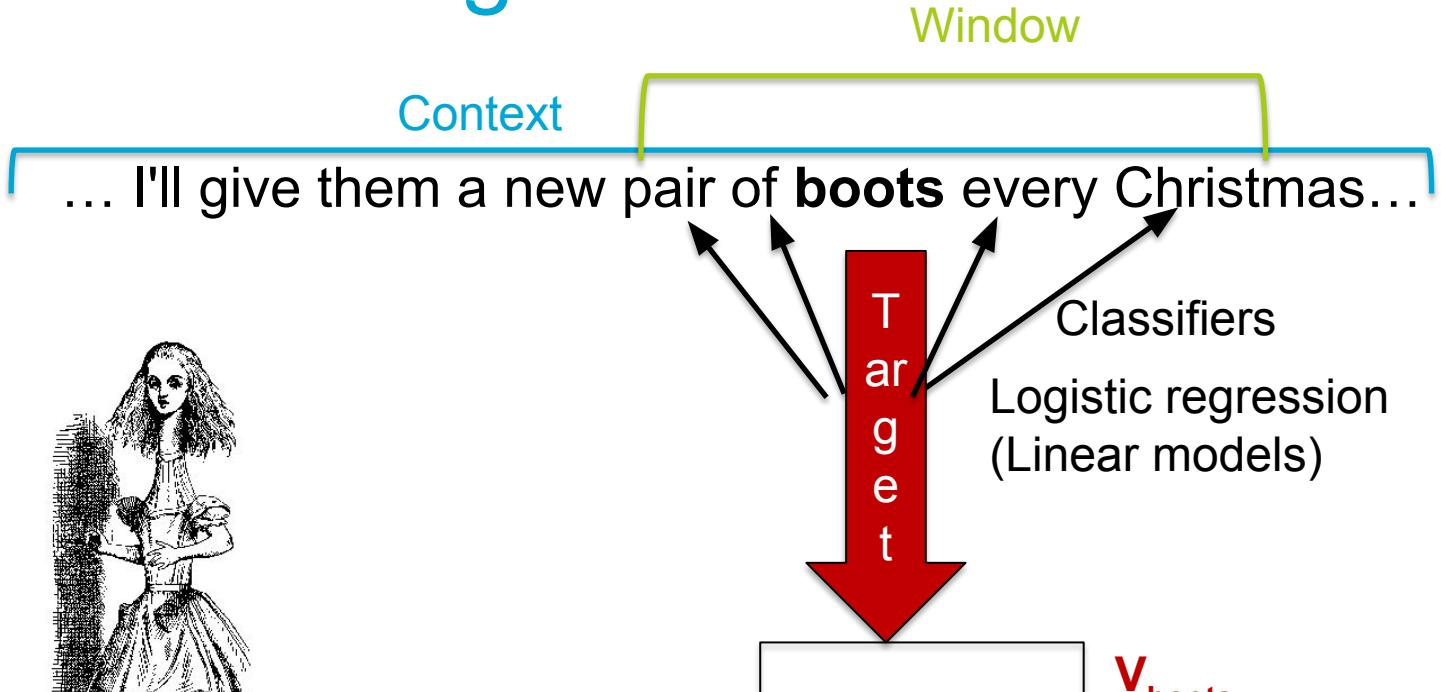
Problems with SVD

- Computational cost scales quadratically for $n \times k$ matrix
- Hard to incorporate new words or documents
- Does not consider order of words

Word2Vec



Embedding overview



word2vec Approach to represent the meaning of word

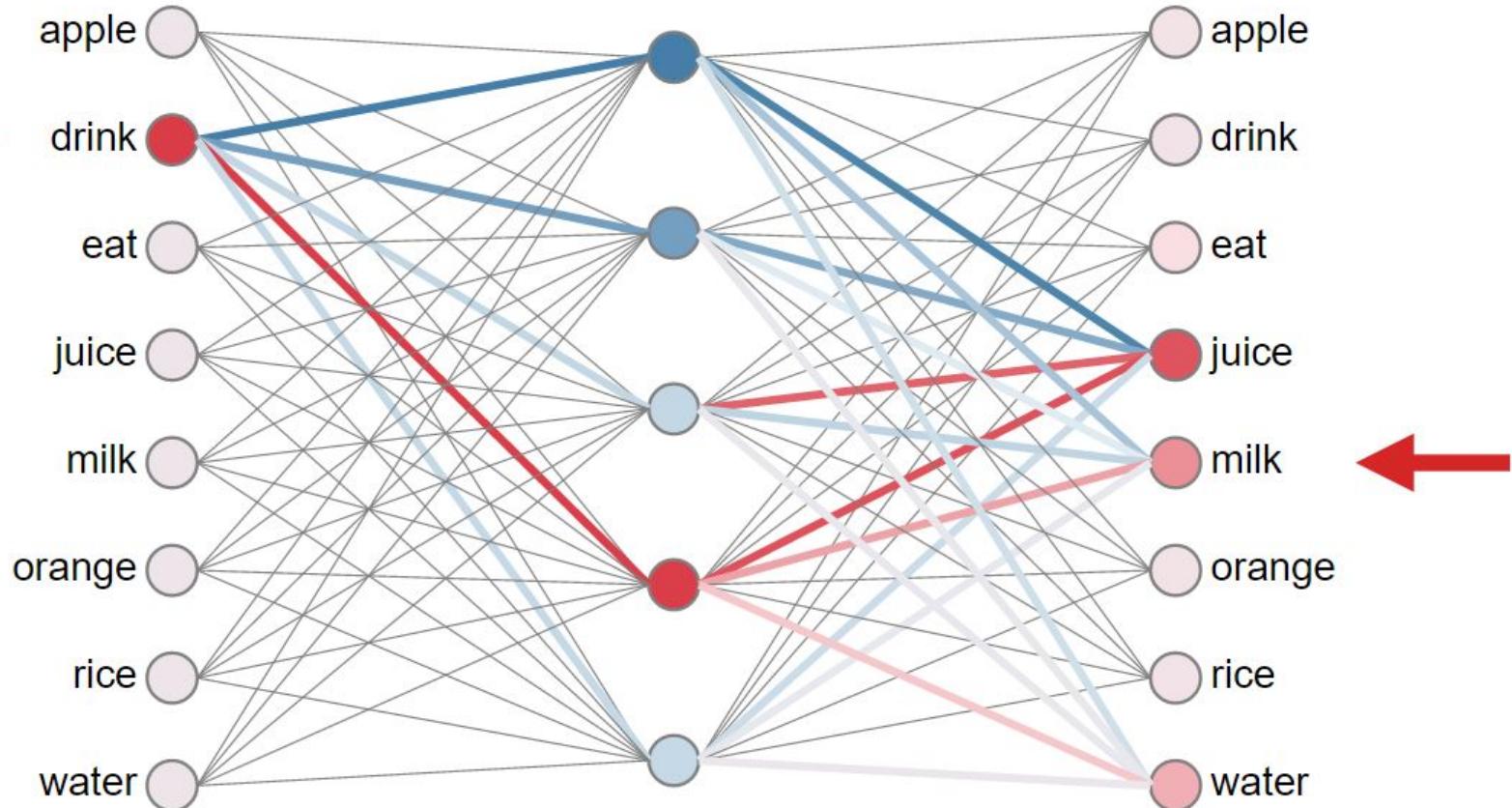
- Represent each word with a low-dimensional vector
- Word similarity = vector similarity
- **Key idea:** Predict surrounding words of every word
- Faster and can easily incorporate a new sentence/document or add a word to the vocabulary

Training Data

1. eat|apple
2. eat|orange
3. eat|rice
4. drink|juice
5. drink|milk
6. orange|juice
7. apple|juice
8. rice|milk

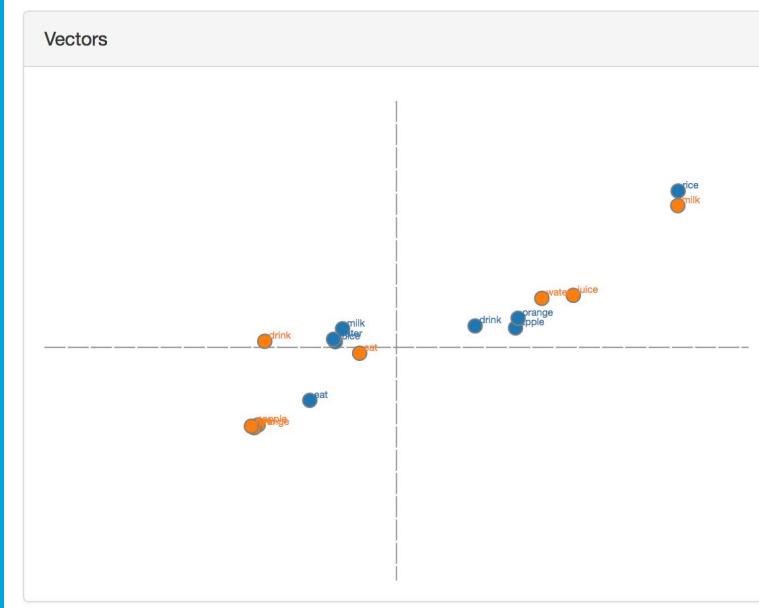
Concept :

1. Milk and Juice are drinks
2. Apples, Oranges and Rice can be eaten
3. Apples and Orange are also juices
4. Rice milk is actually a type of milk!

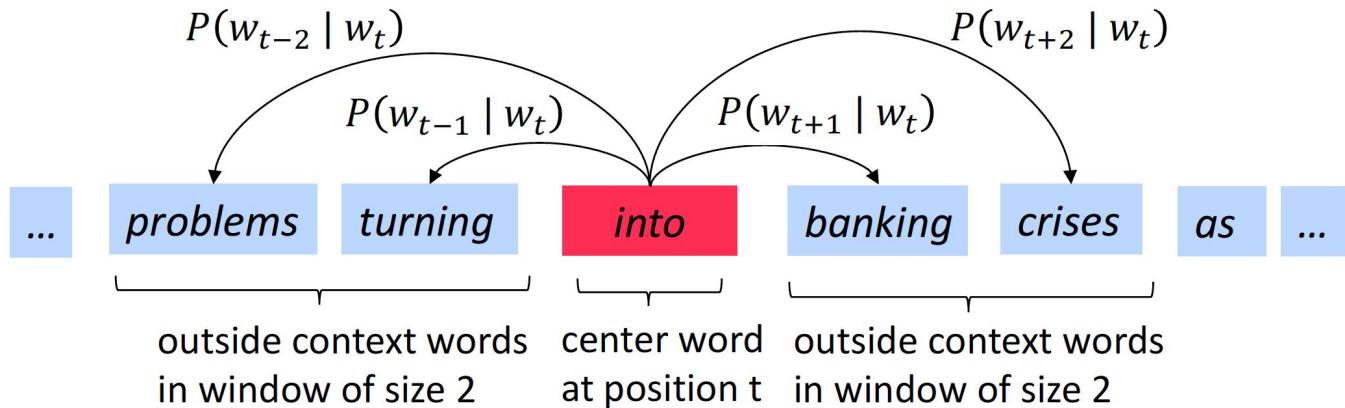


Word Embedding Visualization

<http://ronxin.github.io/wevi/>

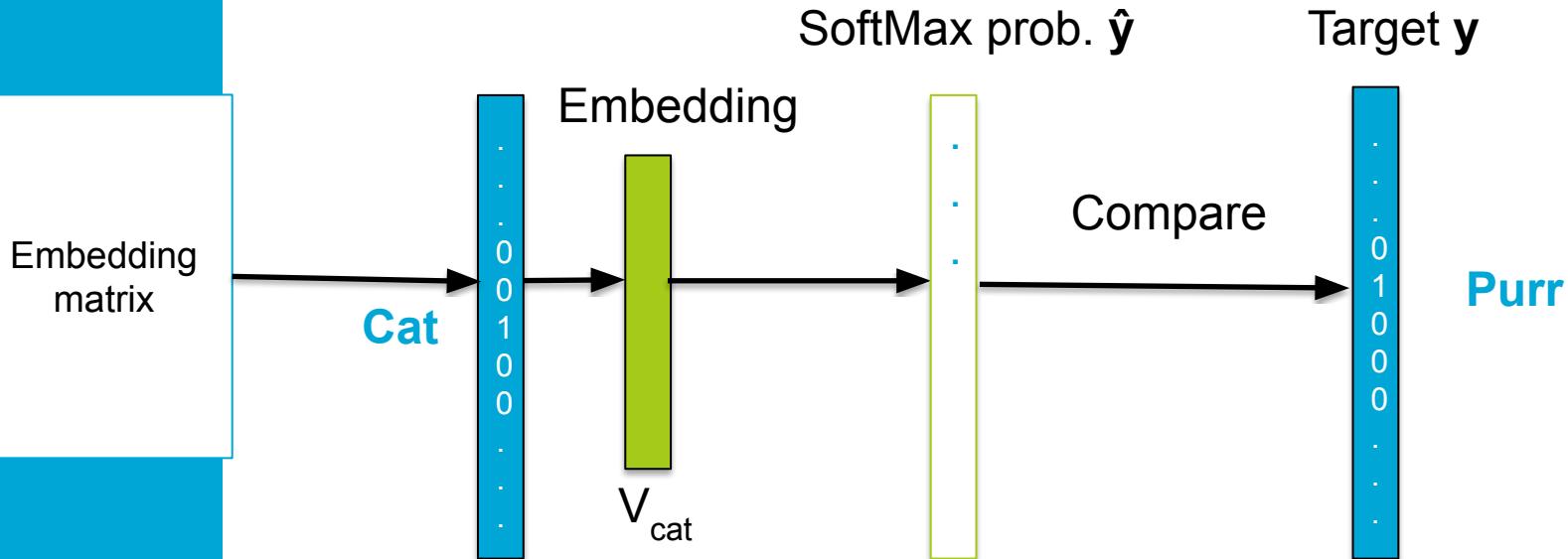


Context window



Context can be anything. E.g., a surrounding n-gram, a randomly sampled set of words from a fixed size window around word....

Architecture



Softmax

W_O - target (purr)

W_I - source (cat)

v' - “output” vector

v - “input” vector

$$p(w_O|w_I) = \frac{\exp\left(v'_{w_O}^\top v_{w_I}\right)}{\sum_{w=1}^W \exp\left(v'_{w_I}^\top v_{w_I}\right)}$$

[0,1]

Source with
other target

Target
output Source
input

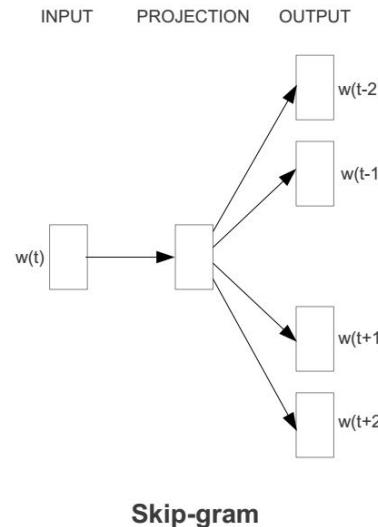
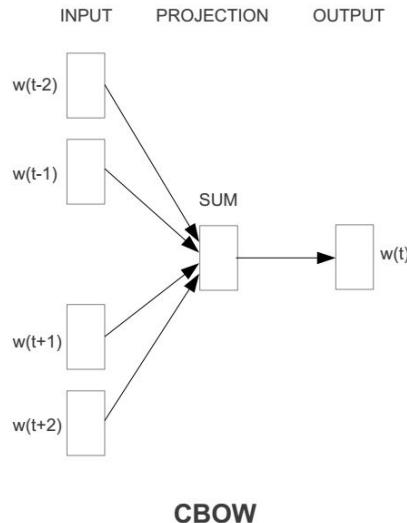
- Minimize error (predicted vs actual) using e.g, gradient descent:
 - $L(\hat{y}, y) = \sum (y_i * \log \hat{y}_i)$

Some other concepts

- Known as neural embedding
- Often optimized using two methods
 1. Hierarchical Softmax
 2. Negative Sampling
- CBOW(continuous bag-of-words) and Skip-gram based training

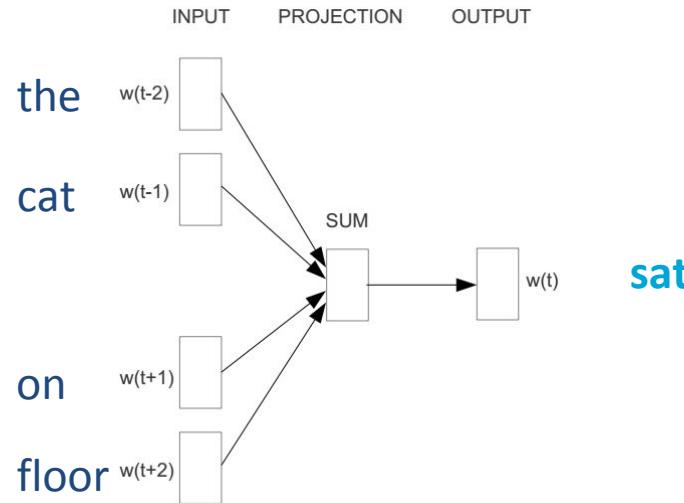
Word2vec - 2 neural models

- **Continuous Bag of Word** (CBOW): use a window of word to predict the middle word
- **Skip-gram** (SG): use a word to predict the surrounding ones in window.



Word2vec – Continuous Bag of Words

- E.g. “The cat sat on floor”
 - Window size = 2



Some other concepts

- Known as neural embedding
- Often optimized using two methods
 1. Hierarchical Softmax
 2. Negative Sampling
- CBOW(continuous bag-of-words) and Skip-gram based training
- **Down Sampling of Frequent words**
- **Phrasal and paragraph vectors**

Questions?



Using word2vec in your research . . .

- Easiest way to use it is via the Gensim library for Python (tends to be slowish, even though it tries to use C optimizations like Cython, NumPy)

<https://radimrehurek.com/gensim/models/word2vec.html>

- Original word2vec C code by Google

<https://code.google.com/archive/p/word2vec/>

Global Vectors for Word Representation (GloVe)

<https://nlp.stanford.edu/projects/glove/>

Main Idea

- Uses ratios of co-occurrence probabilities, rather than the co-occurrence probabilities themselves

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k ice)/P(k steam)$	8.9	8.5×10^{-2}	1.36	0.96

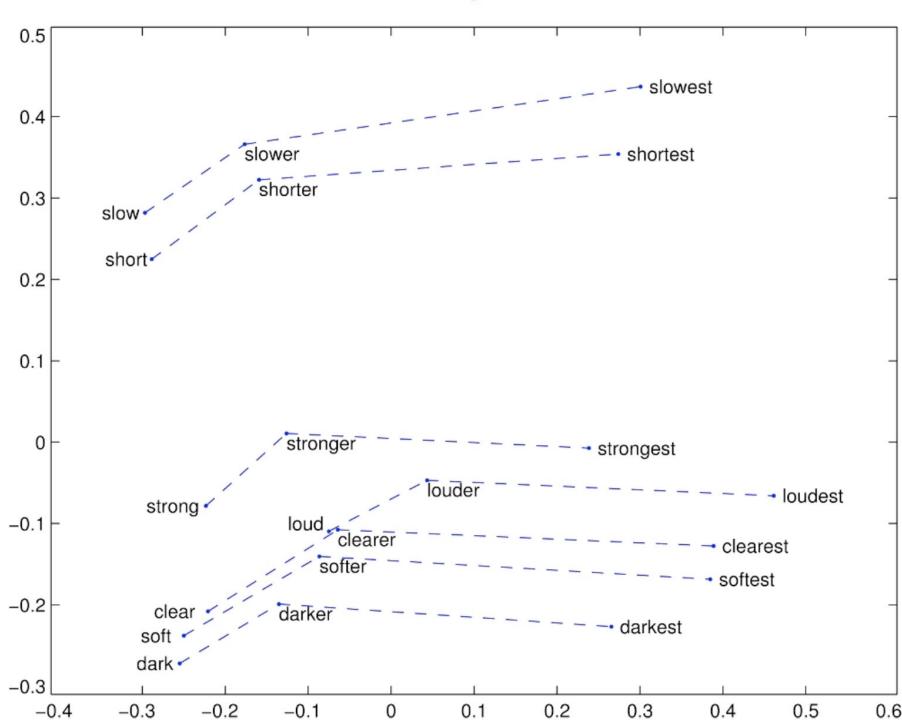
GloVe

- Fast training
- Scalable to huge corpora
- Good performance even with small corpus, and small vectors
- By Pennington, Socher, Manning (2014)

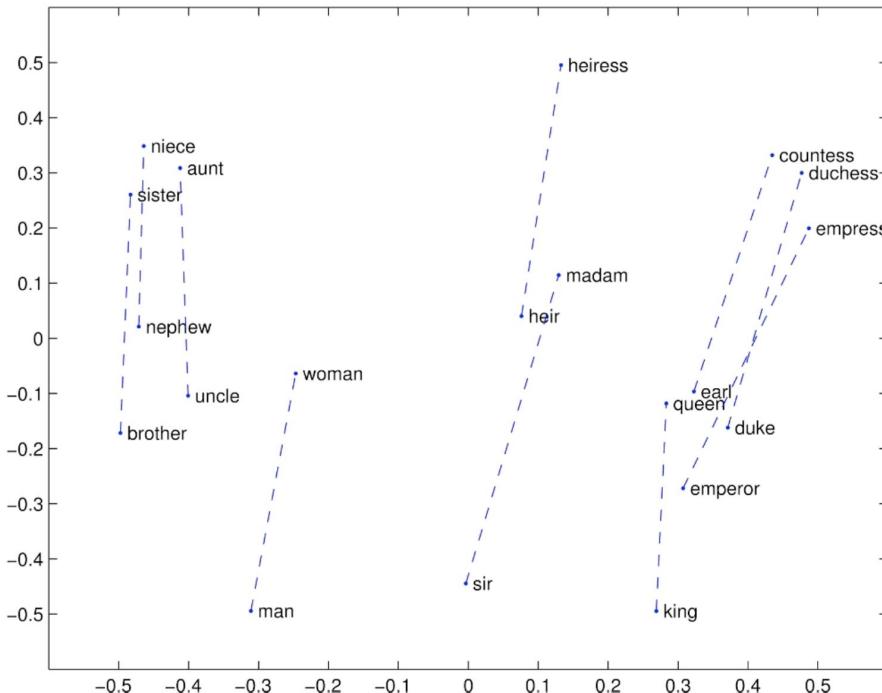
Least Squares Problem

- Often used for data fitting
- Several equations, set models toward a solution
- Solution **minimizes** sum of the squares of the
- **Residual:** the difference between an observed value, and the fitted value provided by a model.

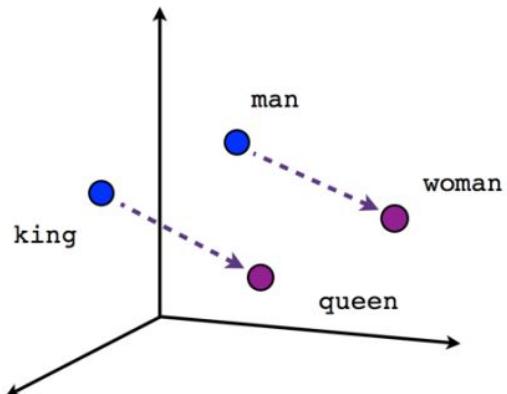
Examples



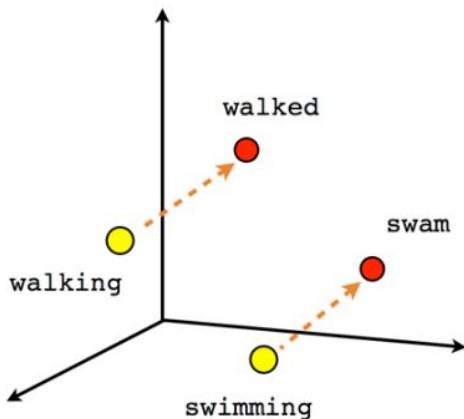
Examples



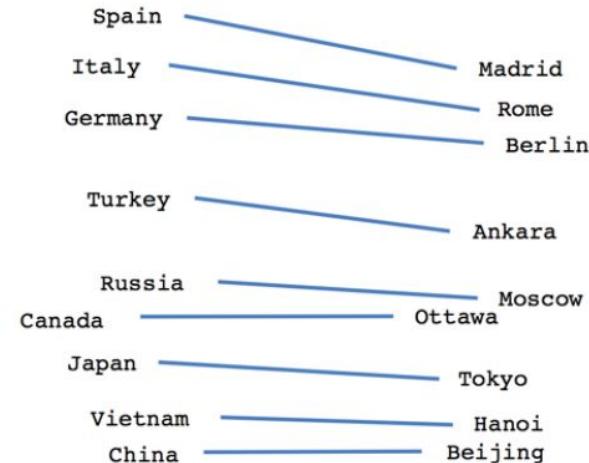
Examples



Male-Female



Verb tense



Country-Capital

$\text{vector[Queen]} = \text{vector[King]} - \text{vector[Man]} + \text{vector[Woman]}$

Word Analogies

Test for linear relationships, examined by Mikolov et al. (2014)

$$a:b :: c:?$$



$$d = \arg \max_x \frac{(w_b - w_a + w_c)^T w_x}{\|w_b - w_a + w_c\|}$$

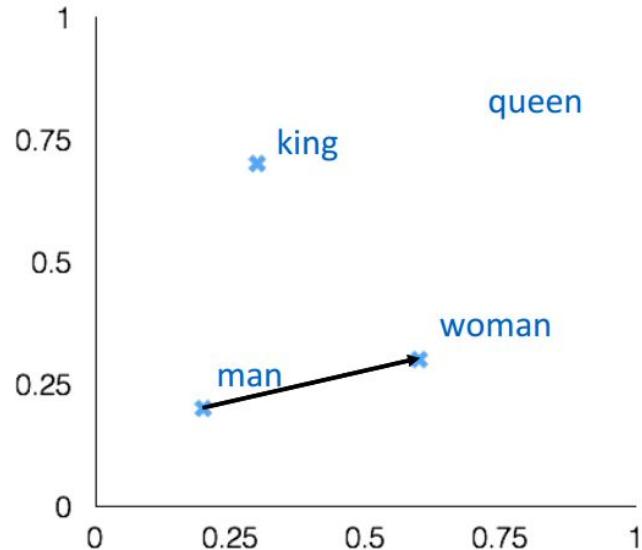
man:woman :: king:?

+ king [0.30 0.70]

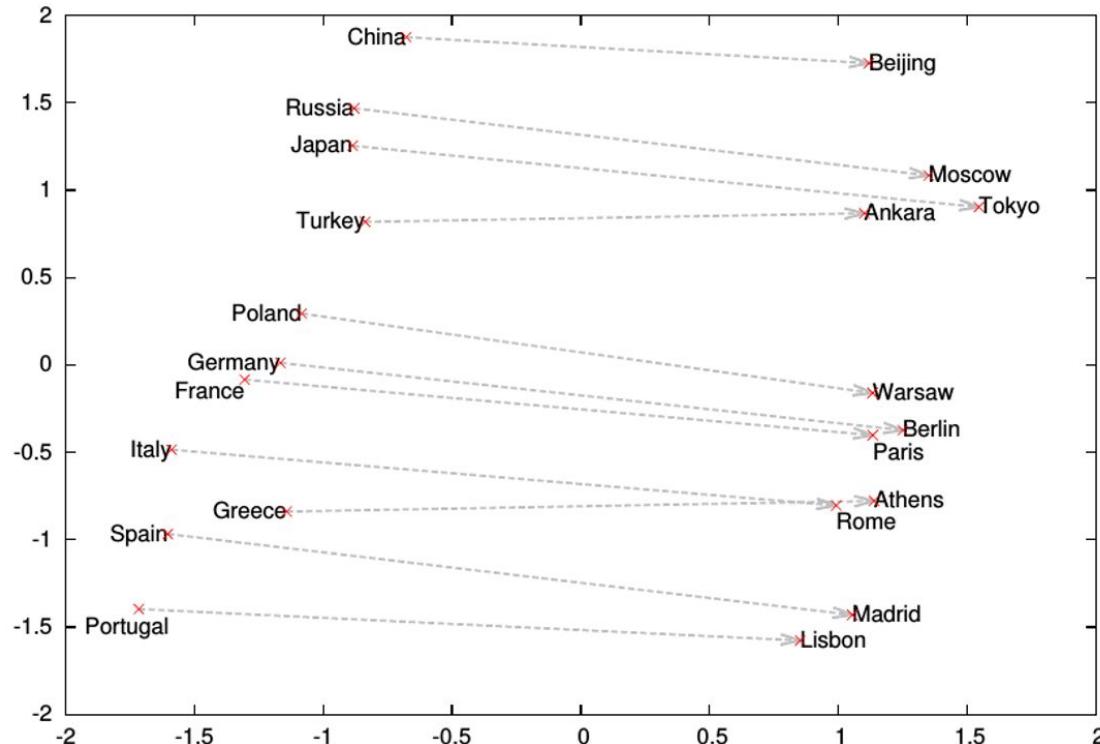
- man [0.20 0.20]

+ woman [0.60 0.30]

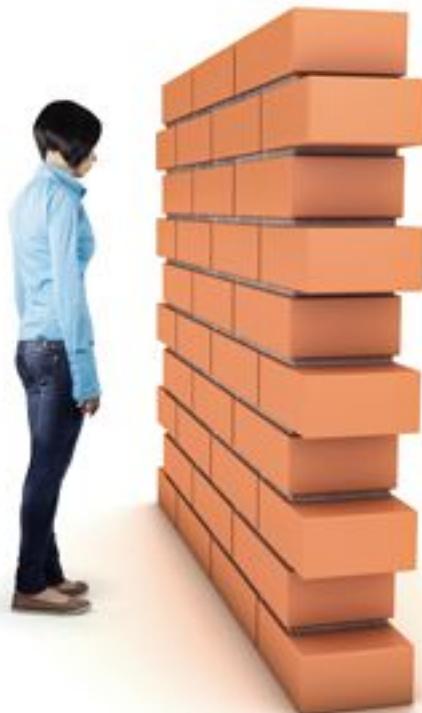
queen [0.70 0.80]



Word analogies



Limitations Embeddings



Limitations 1

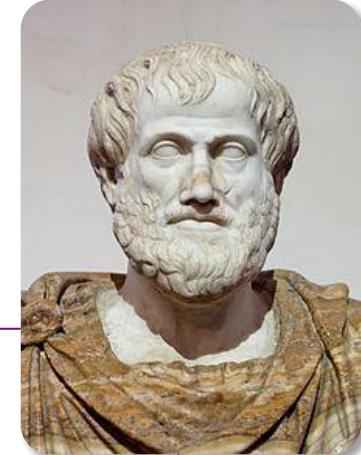
- Very vulnerable, and not a robust concept
- Can take a long time to train
- Non-uniform results
- Hard to understand and visualize

Limitations 2

- Inability to handle unknown or OOV words
- No shared representations at sub-word levels
- Scaling to new languages requires new embedding matrices

Towards Analogy-Enabled Recommendation

- Towards human-level communication...
 - Main tool between humans: **analogies**
 - “Core of human cognition”
 - “The thing that makes us smart”
 - Does this also work for recommendation?
 - **Querying and Explaining**



Analogy (from Greek ἀναλογία, "proportion") is a cognitive process of transferring information or meaning from a particular subject (*the source*) to another particular subject (*the target*), and a linguistic expression corresponding to such a process.[...]



Towards Analogy-Queries

- **Explain complex concepts intuitively**
- Describe concepts **not using the respective domain vocabulary** or domain knowledge
- **Communicate a lot of semantics in only a few words**

Analogy Semantics for Recommendations

- **Completion query:**

$$?: [a_1, a_2] :: [b_1, ?]$$


::

?



Analogy Semantics for Recommendations

- Completion:

$$?: [a_1, a_2] :: [b_1, ?]$$



::



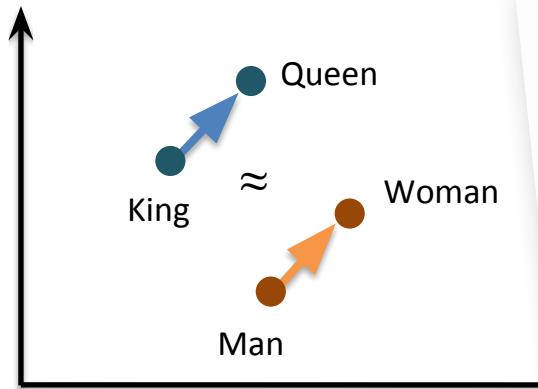
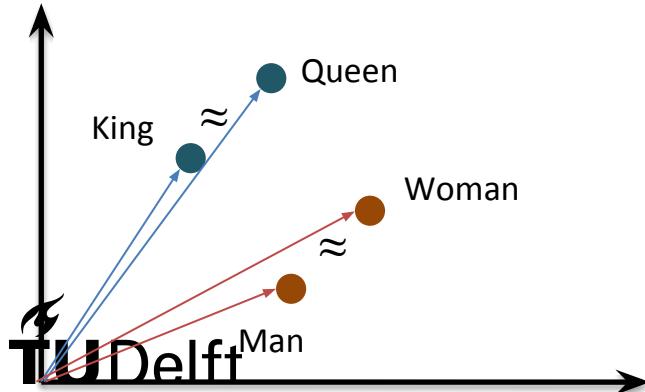
Analogies from an Algorithmic Point of View

- **Multiple Choice:**
- $? : [a_1, a_2] :: ? \{[b_1, b_2], \dots, [z_1, z_2]\}$



Solving Analogy Queries

- **Word Embeddings:** Current peak of distributional semantics research based on Neural Networks (e.g., Word2Vec)
 - Has many surprising applications...
 - Learn a semantic vector representation of words
 - **Claim:** Word Embeddings are good at analogy processing! TRUE?



Distributed Representations of Words and Phrases and their Compositionality

Tomas Mikolov
Google Inc.
Mountain View
mikolov@google.com

Ilya Sutskever
Google Inc.
Mountain View
ilyasus@google.com

Kai Chen
Google Inc.
Mountain View
kai@google.com

Greg Corrado
Google Inc.
Mountain View
gcorrado@google.com

Jeffrey Dean
Google Inc.
Mountain View
jeff@google.com

Abstract

The recently introduced continuous Skip-gram model is an efficient method for learning high-quality distributed vector representations that capture a large number of precise syntactic and semantic word relationships. In this paper we present several innovations that improve both the quality of the vectors and their training speed. By subsampling of frequent words we obtain significant speedup and also learn more regular word representations. We also propose a simple alternative to the hierarchical softmax called negative sampling.

An inherent limitation of word representations is their indifference to word order and their inability to represent idiomatic phrases. For example, the meaning of "Canada" and "Air" cannot be easily captured to obtain "Air Canada". Motivated by this example, we present a simple method for finding phrases in text, and show how our word representations for millions of phrases is possible.

Benchmark Sets

- Word Embeddings are usually evaluated using “weak” analogies, only 25 relationship types: These analogies are near useless!
 - [hot, hotter] :: [fat, fatter] : **Superlative relationship**
 - [Delft, Netherlands] :: [Rome, Italy] : **City-In Relationship**
 - [petal, flower] :: [tire, car] : **Part-Of relationship**



hot hotter hottest



fat fatter fattest



Better Benchmark Sets

SAT®

LEGEND



? legend :: map

- a) subtitle :: translation
- b) bar :: graph
- c) figure: blueprint
- d) key :: chart
- e) footnote :: information

- **SAT analogy challenges**
 - Hand-crafted set used in US-college admission test
 - College students get around 55% of these challenges right
 - Multiple-choice challenges with clear and undisputable “correct” answers
 - 5 choices per challenge
 - 353 challenges
 - Many domains
 - **Difficulty comes from using rare words**

AGS Benchmark Set

- **Multi-Step Crowdsourcing Process**
 - **Multiple-Steps == Easier quality control**
 - Start with **seed word pairs** which are “somehow” related
 - Mined from SimLex-999 or WordSim-353
 - Used **crowdsourcing** to classify each seed word pair into one of 14 categories
 - Based on the categories, **generate new word pairs** sharing the same relationship
 - (Good examples, but also bad examples)
 - Obtain **aggregated analogy ratings** for each pair



The AGL Testset - Domains

Categories	#chall.	#analog	Categories	#chall.	#analog.
Animals /Plants	10	128	House/Furniture/Clothing	17	169
Automobiles/Transportation	5	41	Humans/Human relations	3	28
Electrical/Electronics	4	42	Medicine/Healthcare	3	28
English grammar	11	121	Movies/Music	4	61
Food/Beverages	11	92	People/Profession	12	151
Geography/Architecture	9	144	Sports	4	33

Our AGS-2 Benchmark Set

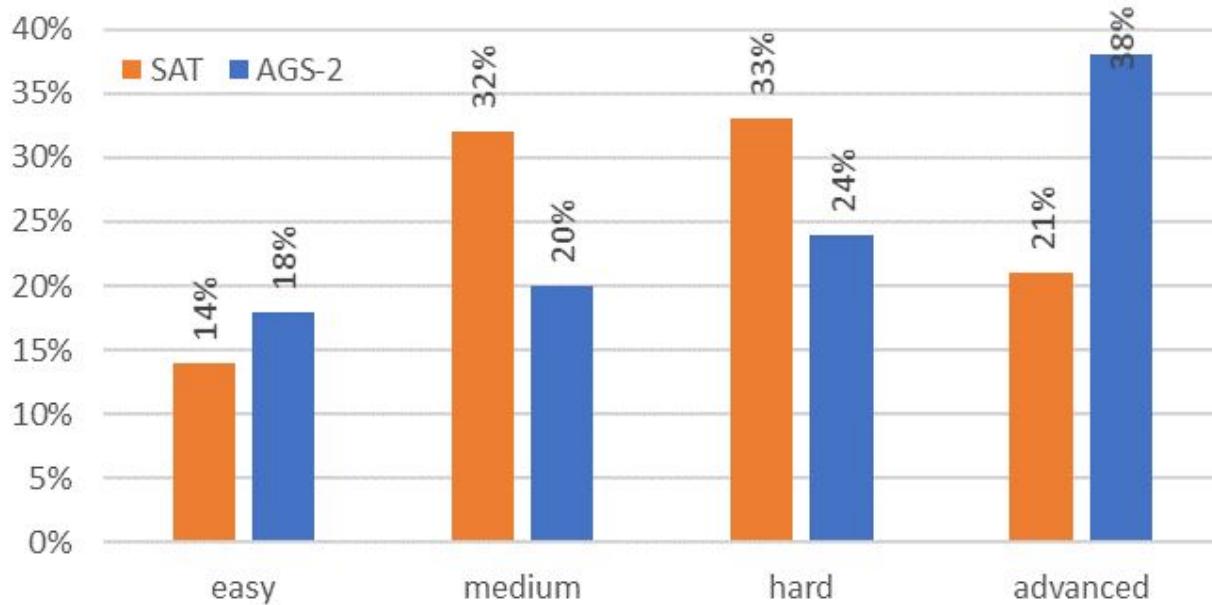
- New concept: **Analogy Rating**
 - Based the definiteness of the analogy, and its overall “quality”
 - **This is subjective, needs to be based on human judgement!**
 - Good example: [Bordeaux Area, France] :: [Napa Valley, US] : **famous-for-good-wine relationship**
 - **Subjective: Use crowdsourcing consensus**



Our AGS-2 Benchmark Set

- New concept: **Difficulty**
 - Analogies imply a reasoning process, which may be easier or harder to follow
 - Difficulty depends on abstractness of reasoning, rareness of vocabulary
 - Again: Subjective, Crowdsourcing consensus necessary

Difficulty of challenges in Analogy datasets



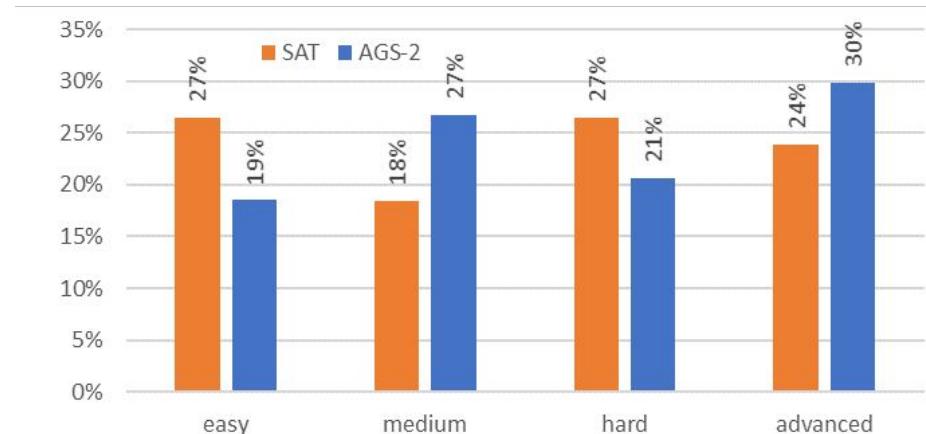


Our AGS-2 Benchmark Set

- Design goal of new AGS Benchmark set
 - **Include a wide variety of analogy semantics**
 - Cover different relationship types
 - Cover some defining analogies, but also some less defining ones
 - Allow for benchmarking all three query types
 - Confirmation, Completion, Multiple Choice
 - **Include analogy rating, difficulty rating** judging the quality of analogon combinations
 - Include analogon with high analogy rating, but also include some with a low rating
 - **Judgments by humans**

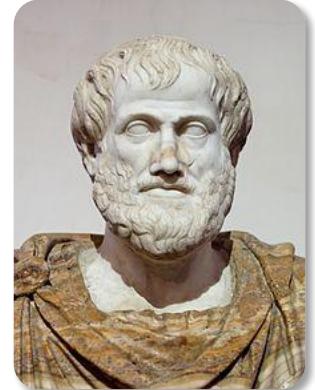
Word-Embeddings: How well do they do?

- Setup: Word2Vec using Wikipedia
- Multiple Choice: Find the best answer from given collection
- Result: Accuracy between 18-30%
 - Word2Vec does not perform well for easy challenges (humans have $\geq 80\%$)
 - Better for hard ones (humans $\leq 20\%$)



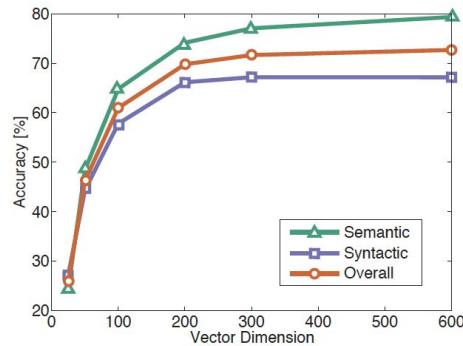
Summary and Outlook

- Human-Centered Information Systems are a big challenge for the future
 - Analogies are likely a central part
 - But: Analogies are hard to process, and hard to evaluate
 - Lot of subjectivity and common-sense consensus
 - Current benchmark datasets are not suitable
- Our AGS dataset is large step towards more semantic evaluation
 - Still, many improvements can be made...
 - Improve dataset covering more aspects of analogies
 - Better and stronger classification of challenges
 - Evaluations with a wider selection of approaches
- Toward analogy-based explanations!

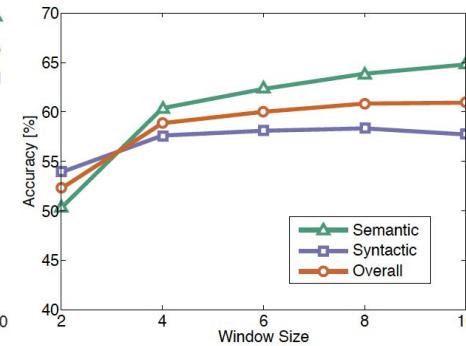


Data limitations

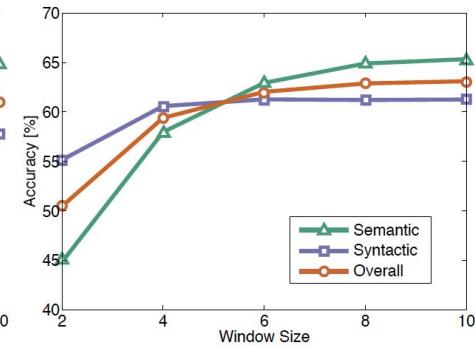
- Asymmetric context (only words to the left) are not as good



(a) Symmetric context



(b) Symmetric context

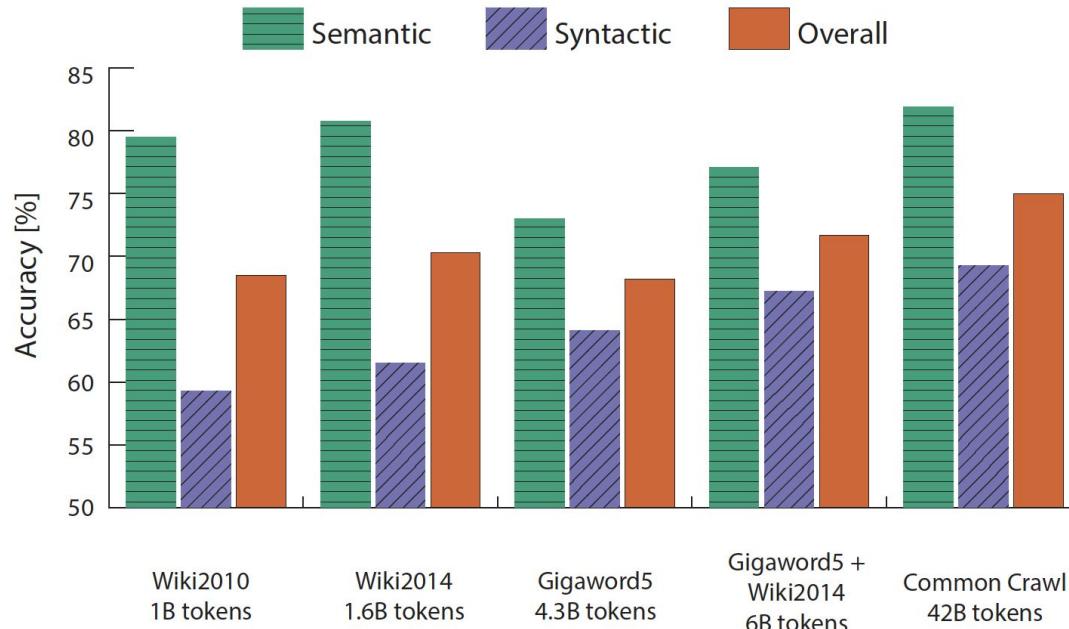


(c) Asymmetric context

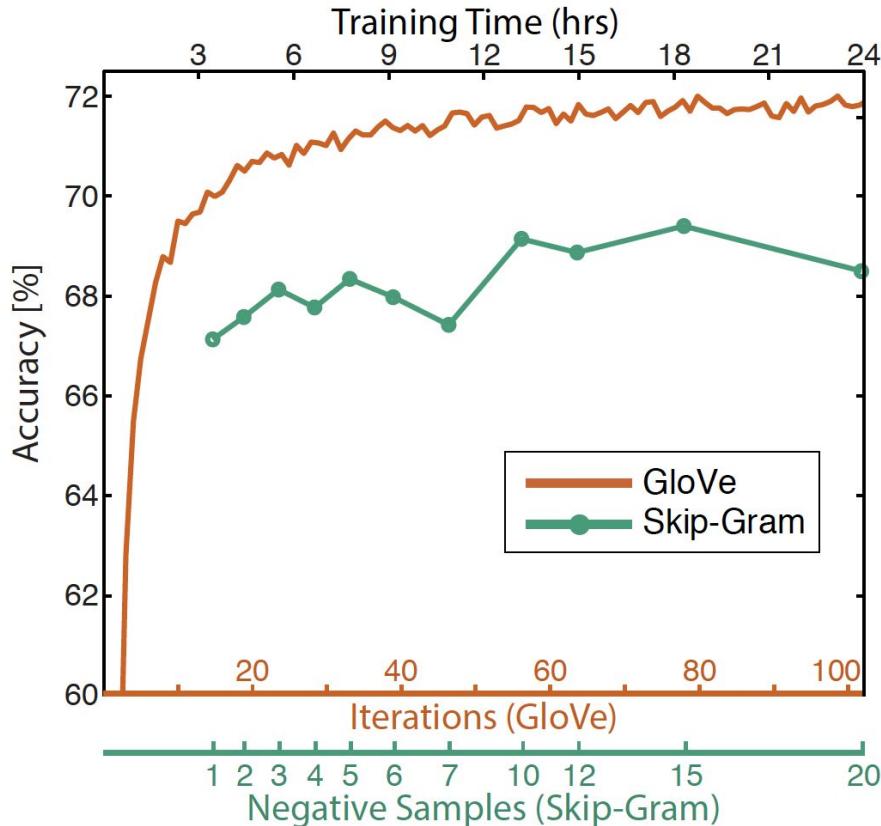
- Best dimensions ~ 300 , slight drop-off afterwards
- But this might be different for downstream tasks!
- Window size of 8 around each center word is good for Glove vectors

More data!

- More data helps, Wikipedia is better than news text!



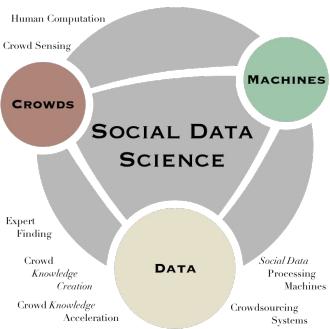
More training!



Today

- Why word embeddings
- Word Embeddings
 - Relation to other dimension reduction, e.g., SVD
 - Word2Vec
 - Skipgrams
 - Continuous Bag of Words (CBOW)
 - GLOVE
- Example: Analogies
 - Limitations
 - Data limitations

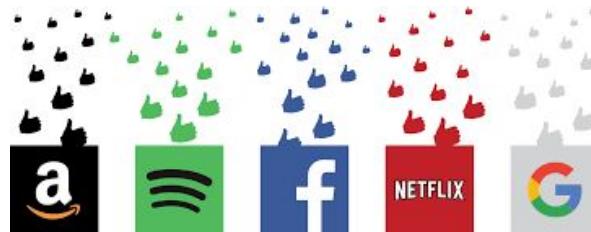
If you liked this course you'll also like...



CS4145 - Crowd Computing

How large groups of people can solve complex tasks that are currently beyond the capabilities of AI, and that cannot be solved by a single person alone.

Gamification, conversational systems, human computation, and much much more...



HUMAN AIDED CHATBOT FOR CAMPUS INFORMATION RETRIEVAL

Background: A new emerging hybrid chatbot system has spawned using human-based computation to improve response flexibility and request comprehension of chatbots.

- In this project, we will test our human aided chatbot based on telegram. Students will use this chatbot to request or provide campus information. Through this project, we will find an effective user interface to improve the quality of results provided by workers. We'll also compare the effectiveness and efficiency of traditional chatbot, hybrid chatbot and webpage-based crowdsourcing platform, by conducting several experiments among students.
-



Wednesday, June 13, 2018

3:35:33 PM

SH Sharad
/start

MY MyCI
Hello Sharad, my name is MyCI. I am a chatbot that can help you find recommendations around campus with the help of other students.
/menu to get the menu

3:35:35 PM

What courses would you like to add to your preference list?

3:35:35 PM

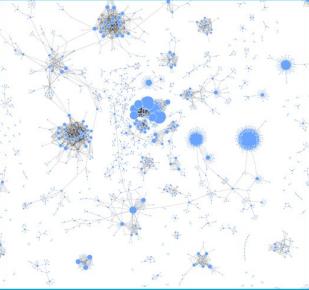


Trasherhunt

Background: Crowdsensing based on street view means the way to collect data by a group of people using web-based street panorama viewer.

In this project, we will test our crowdsensing platforms based on street panorama viewers (panorama data comes from Google and Amsterdam Data Science). Students are supposed to use this platform to find dirty/clean spots and trash bins in Noord-Zuid Lijn area (in Amsterdam). We will then compare the trash bin information collected by our platforms (Google and ADS) with OpenStreetMap (and ADS map). We'll also analyse the relationship between "cleanliness/dirtiness" and the density of trash bins.

RBUTR

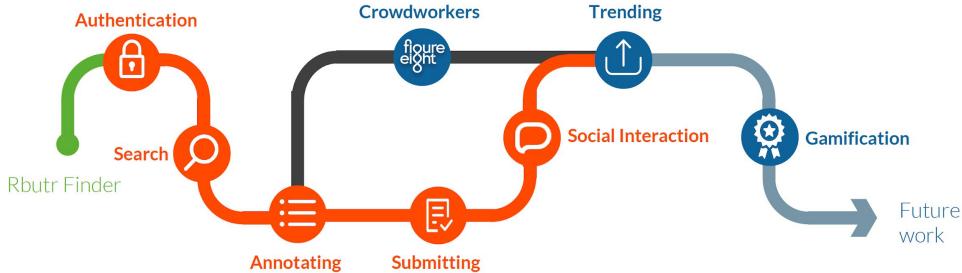


- Created 2012
- 2K active users per week
- 60K total users
- 28K edges, 40K nodes

Encourage users to annotate relationship between nodes:

E.g., rebuttal, disagree, irrelevant

Use this information for an information seeking task.



Related URL Search

<https://www.vox.com/2016/8/23/12608316/epipen-price-my>

Search

Showing 15 results.

<https://np.reddit.com/r/WikiTextBot/wiki/index>

SOURCES :

https://www.reddit.com/r/TrueReddit/comments/6jklq2/epipens_400_percent_price/,
https://www.reddit.com/r/TrueReddit/comments/6jklq2/epipens_400_percent_price/,
https://www.reddit.com/r/TrueReddit/comments/6jklq2/epipens_400_percent_price/

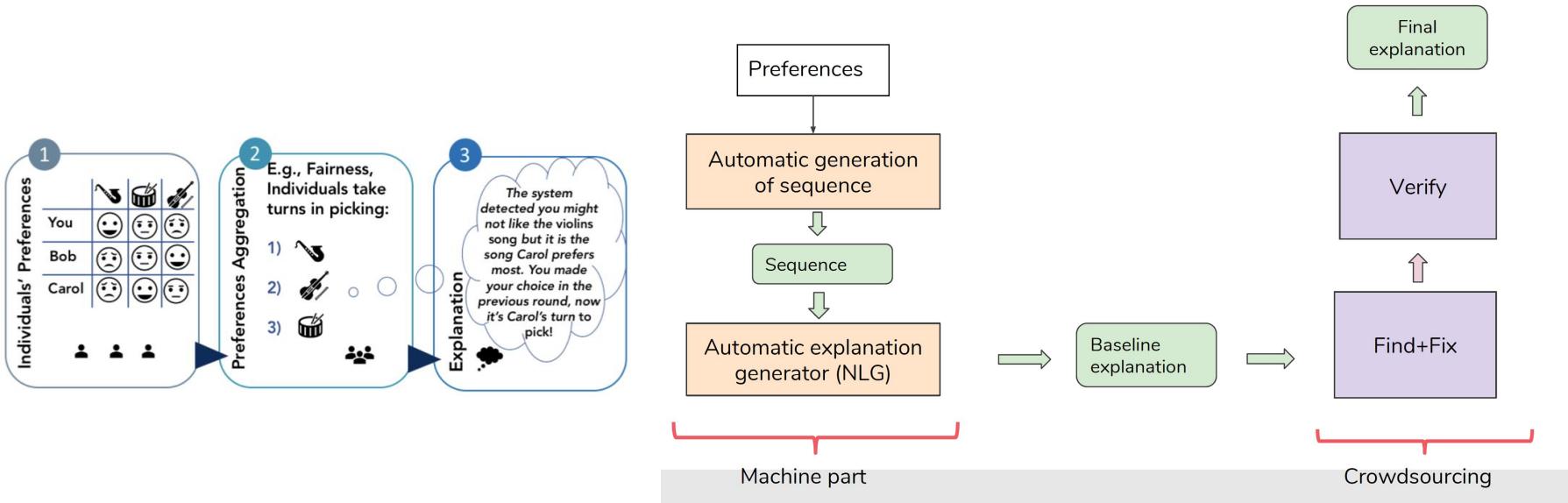
Rebuttal : 2

Contrary : 0

Irrelevant : 1

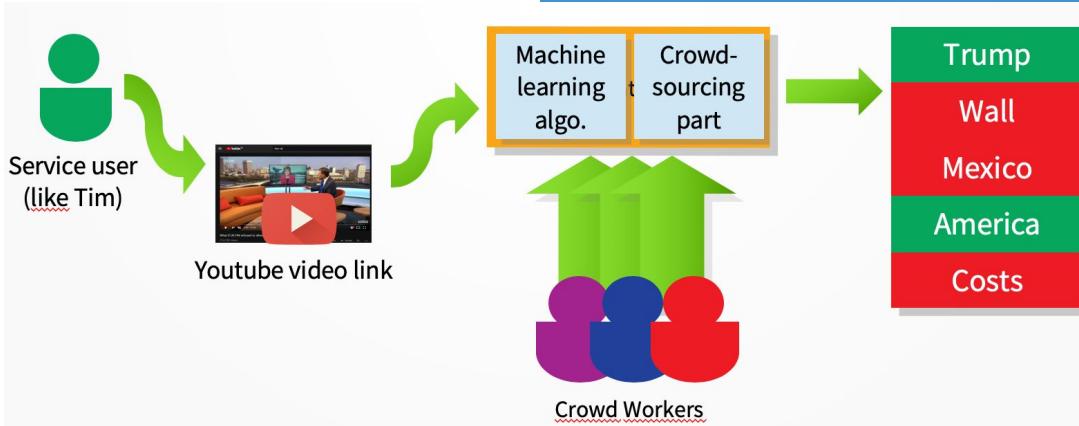
Add/View comments

EXPLAINING SEQUENCES OF RECOMMENDATIONS



Use human computation to generate and evaluate explanations.

CAPTURING BIAS IN VISUAL NEWS SOURCES



Background: Using experts to harness ground truth can be tedious, time consuming and expensive. That is why with the help of crowdsourcing platforms and computer interfaces we could train a crowd for task that we would normally need several experts for. Additionally we could assist experts with their job.

In this project we want to help humanities and social science experts to codify videos with the help of the crowd. To be able to help the experts we will need to be able to cut the videos into smaller chunks and assign them to the Crowd. A useful library and tool to cut a complex task into simpler ones could be CrowdForge from Google Research. We will experiment with a specific example on coding bias with the use of an expert example and will scale it up with the use of the crowd in Figure8 platform.

Next up

Interviews 12 and 13th of March

P4: Reidsma, Dennis. "Exploiting 'subjective' annotations." Proceedings of the Workshop on Human Judgements in Computational Linguistics. Association for Computational Linguistics, 2008.

Core IR starts next week with Claudia Hauff!

Questions?



Credits

- Natural Language Processing Lab, Texas A&M University
- Vagelis Hristidis, prepared with the help of Nhat Le
- Richard Socher, Stanford CS224d: Deep Learning for NLP
- Paper reference:
 - Christoph Lofi and Nava Tintarev. "Towards analogy-based recommendation: benchmarking of perceived analogy semantics". In *Complex-Rec workshop at Recsys'17*. 2017