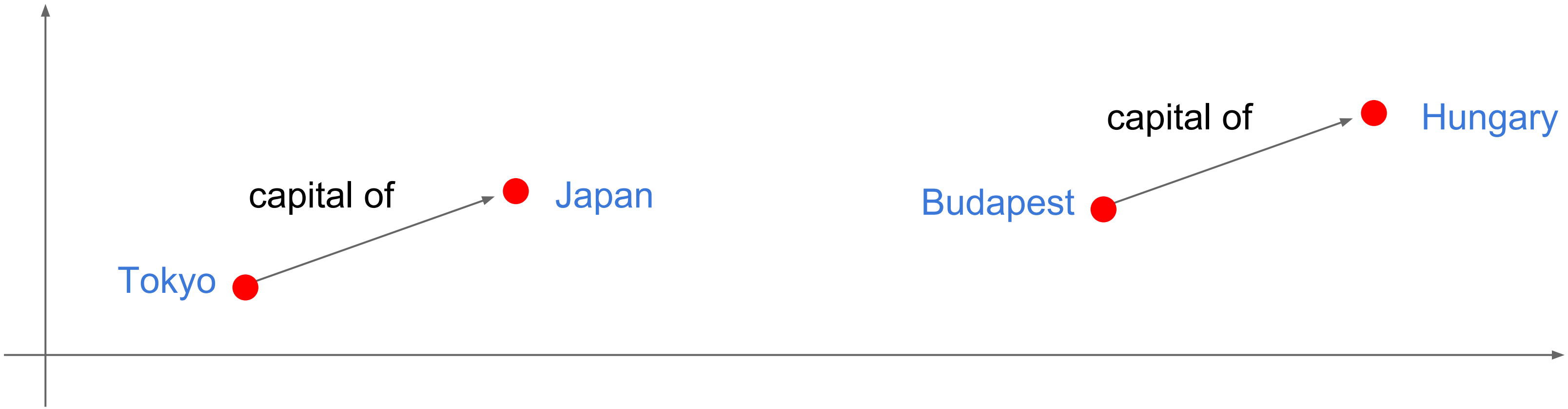
Text embeddings

Sep 26, 2017

Fujitsu’s demonstration

- On Furukawa AI conference in Yokohama on July 6, 2017 Fujitsu engineers demonstrated a poster about a search and topic modelling engine
- Features
  - Takes all company documents (internal reports, marketing materials, emails, patents, ...)
  - Measures similarity of documents
  - Creates a map of documents with groups and distances between the docs and their groups
- Use cases
  - Knowledge management: Identify and connect groups in the company who might have relevant knowledge for each other
  - Search engine: documents become searchable based on similarity to given query
- Drawbacks
  - Pre-trained - not adapted to the specifics of the company
  - Does not support English

Remember the key idea: Vector representation of text



- Words can be mapped to vectors, so math works on text:

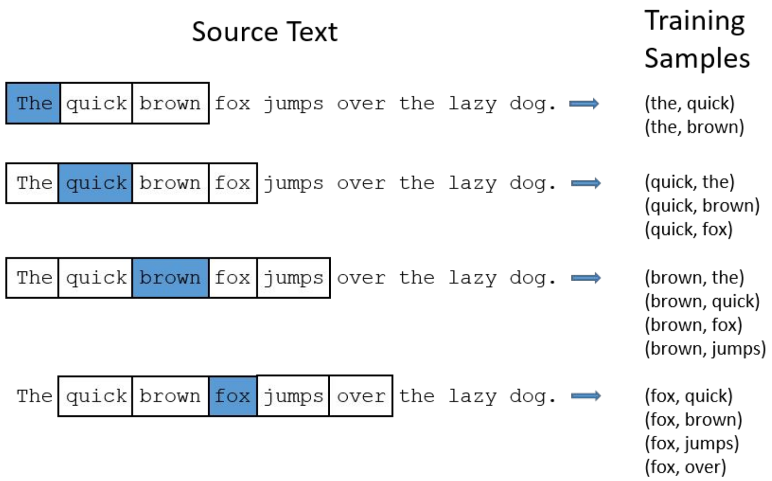
Father - Man + Woman = Mother

Trump - USA + Japan = Abe

Cathode - Negative + Positive = Anode

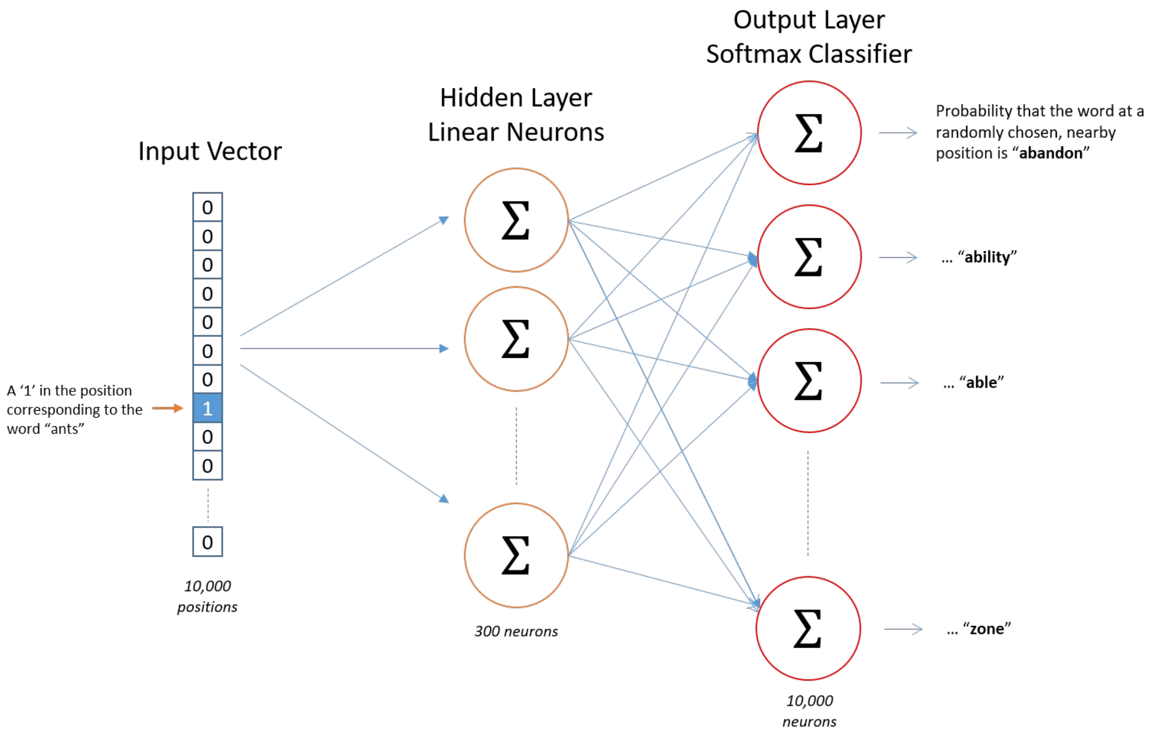
Skip-gram model

- Given a word in the middle of a sentence, look at the words nearby and pick one at random
- Train a network to tell us the probability for every word in the vocabulary of being “nearby”
  - Nearby == Window size
  - Window size = 10 means 5 ahead + 5 behind
- Train the neural network by feeding it word pairs found in training documents



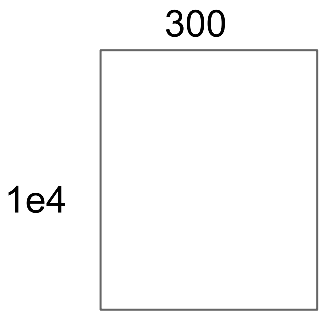
Architecture

- Assume vocabulary of 10000 words
- One-hot encode the input
- Hidden layer has 300 neurons
- Softmax output with 10000 neurons
- Reminds of an autoencoder



Word vectors

- == hidden layer state for every one-hot vector
  - == hidden layer input weights
  - embedding space
- The output of the network is unimportant after the training is done. We needed only the hidden layer weights
- If two different words typically have similar “contexts”
  - then the output of the softmax is similar
  - so they must have similar word vectors
  - they will be close in the embedding space



FETI-model

With implementation details

Why do we want a different model?

- We want to be specific to the Li-ion battery domain
  - == we want to be good there and only there
  - no need to know about philanthropy, guerilla warfare or gardening
- We have very limited training samples
  - only a few thousand abstracts
- We want to know
  - what the paper is about
  - What papers are similar
- So we decided to create different embedding space

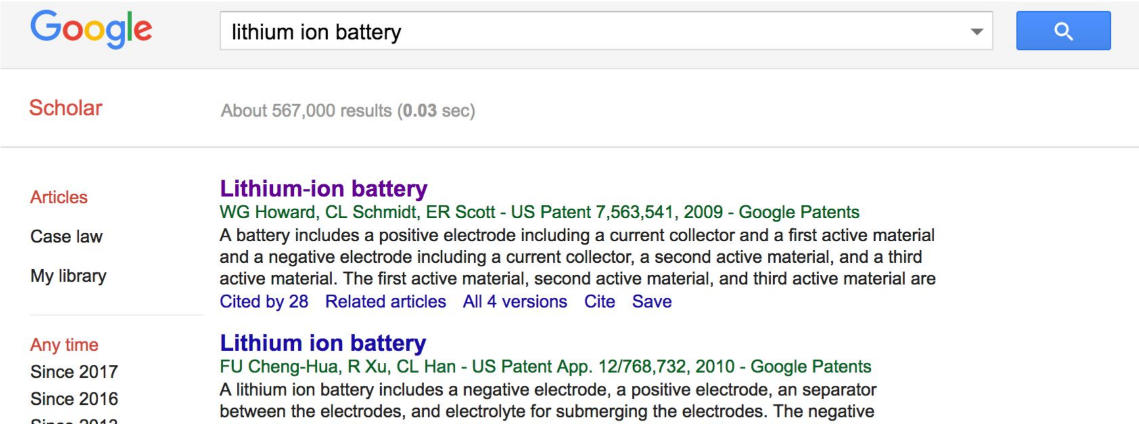
Steps

1. Collect papers by web crawler
   - Google Scholar gives only title, authors, year + 3 line abstract
2. Collect full abstracts
   - Script that downloads full abstracts from 20 different websites
3. Create and clean dictionary
   - Identify and count important words for LIBs
4. Train neural network
   - Learn relationships between words
5. Evaluate documents
   - Identify relationships between documents

Step 1: Collect papers by web-crawler

- Developed a script that Searches Google Scholar for LIBs

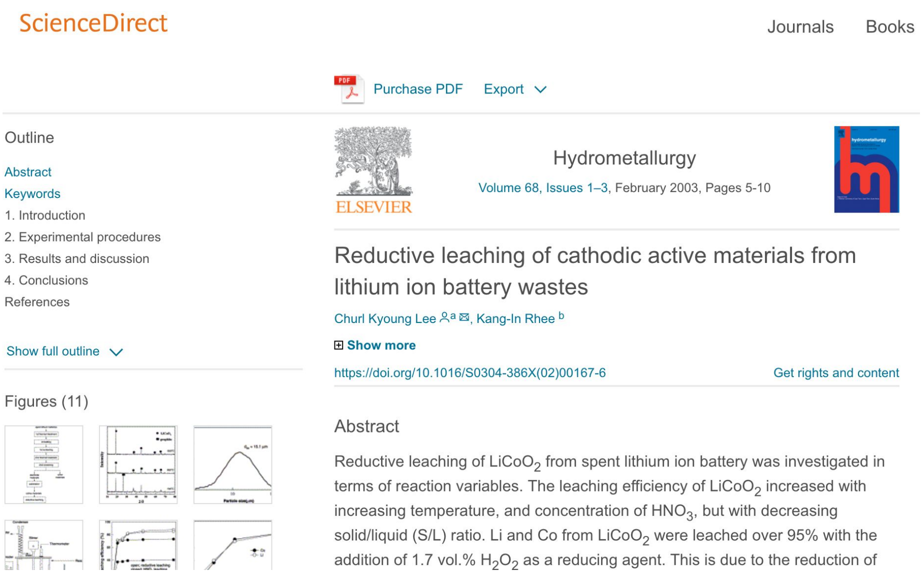
- Downloads metadata
+ 3 lines from the abstract



- Collected 5838 papers from 2009 to 2017

Step 2: Collect full abstracts

- The full abstracts must be downloaded one-by-one with the following algorithm:
    - go to website of paper
    - find 3 lines given by Scholar
    - download all other lines



- Full abstract contains the most important information

Step 3: Create and clean dictionary

- The whole corpus contains 815 394 words (32 064 different words)
- Many of these elements are not important:

LiFePO4/C of **high purity grade** was successfully **synthesized** by **microwave accelerated sol–gel synthesis** and showed **excellent electrochemical performance** in terms of specific **capacity** and **stability**. This **cathode material** was characterized in **battery configuration** with a **graphite counter electrode** by **USABC–DOE tests** for power-assist **hybrid electric vehicle**. It yielded a non-conventional Ragone plot that represents **complexity of battery functioning** in **power-assist HEV** and shows that the **pulse power capability** and available **energy** of such a **battery** surpasses the **DOE** goal for such an application.

- The system should not learn the grammar and general expressions, because we do not need a full language model (here!)

Step 3: Create and clean dictionary

- Most frequent words are grammatically important, but contain no information regarding the specific field

- We have to identify those words which are significant in our domain

Top of the original dictionary with the number of occurrences

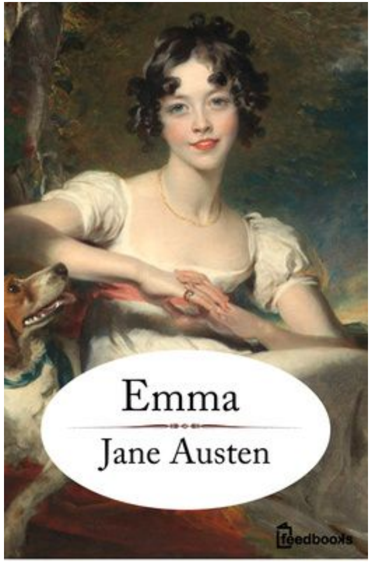
('the', 53971)
('of', 33299)
('and', 27803)
('a', 26294)
('to', 13502)
('in', 13142)
('battery', 9868)
('for', 9397)
('is', 9049)
('with', 8341)
('capacity', 6908)
('by', 6335)
('at', 6327)
('that', 5661)
('material', 5593)
('are', 4923)
('high', 4544)
('performance', 4459)
('lithium', 4386)

Step 3: Create and clean dictionary

- Find a corpus which contains common words and the same grammar, but contains no information about our domain:

  Jane Austen: Emma (written in 1815)

- The first practical battery (Daniell cell was invented in 1836)



- By subtracting this corpus we can remove the non-relevant information
  - words were added back manually to the dictionary: *positive, negative, material, stability, property, performance*

- The cleaned dictionary with contains 27506 different words (only 4558 words were removed)

Step 3: Create and clean dictionary

- The filtered dictionary contains the most relevant words of the domain

- The words/expressions are identified but no context meaning was associated with these words

→ Find meaningful expressions

Top of the filtered dictionary with the number of occurrences

('battery', 9868)
('capacity', 6908)
('material', 5593)
('performance', 4459)
('lithium', 4386)
('electrode', 4219)
('electrochemical', 3991)
('anode', 3670)
'cycle', 3548)
('ion', 3025)
('cell', 2925)
('electrolyte', 2782)
('structure', 2474)
('composite', 2360)
('energy', 2248)
('cycling', 2118)
('cathode', 2100)
('current', 2081)
('discharge', 2012)
('temperature', 1942)

Step 3: Create and clean dictionary

- We do not want to distinguish *battery* from *batteries*
  - stemming the words (finding the roots) is important

- Certain words can always appear together in a phrase
  - Word combination can have a different meaning as the separated words. Chunks/expressions were identified to be handled jointly

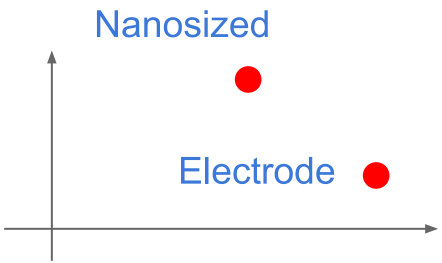
- E.g.: *"Electric vehicle"* and *"electric conductivity"* has different meanings and vehicle and conductivity are not necessarily related

- 75618 different chunks were identified

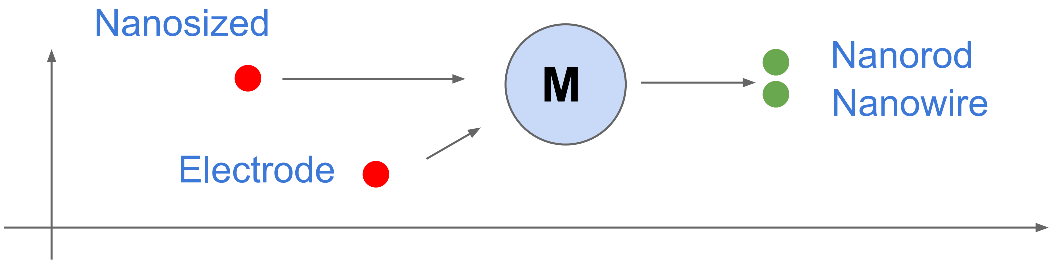
Most frequently appearing chunks

('lithium-ion battery', 1318)
('this paper', 654)
('current density', 597)
('anode material', 564)
('electrochemical performance', 554)
('this work', 426)
('reversible capacity', 391)
('discharge capacity', 374)
('capacity retention', 350)
('cathode material', 292)
('electric vehicle', 249)
('this study', 249)
('high capacity', 246)
('rate capability', 244)
('electrochemical property', 228)
('specific capacity', 224)
('active material', 221)

Step 4: Train neural network

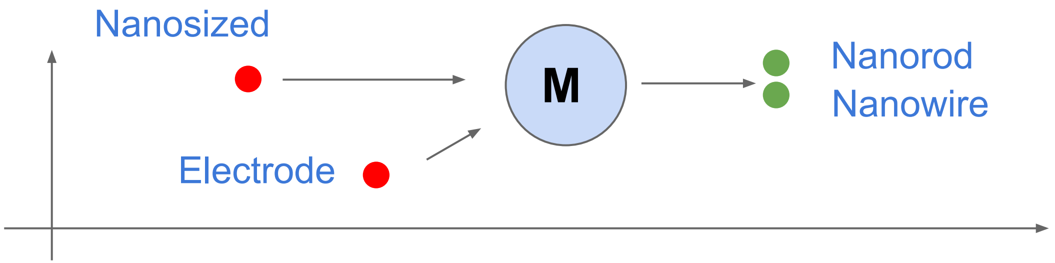


- We want to create the vector space of the chunks (words)
- Key idea: “Two words should be **close** if they show up in **similar context**“
- Context = 2 words from the abstract
  - E.g.: Nanosized and Electrode
- Our Model **M** tries to predict a 3rd word from the same abstract
  - E.g.: Nanorod or Nanowire



Step 4: Train neural network

- "Nanorod" and "Nanowire" are similar, because they show up in similar contexts



- "Nanorod" and "Liquid" will be far, because they show up in different contexts

- We have to solve 2 tasks together:

1. Find optimal position for every word in the space (=EMBEDDING)
2. Find the weights of the neural network M that correctly predicts words from context

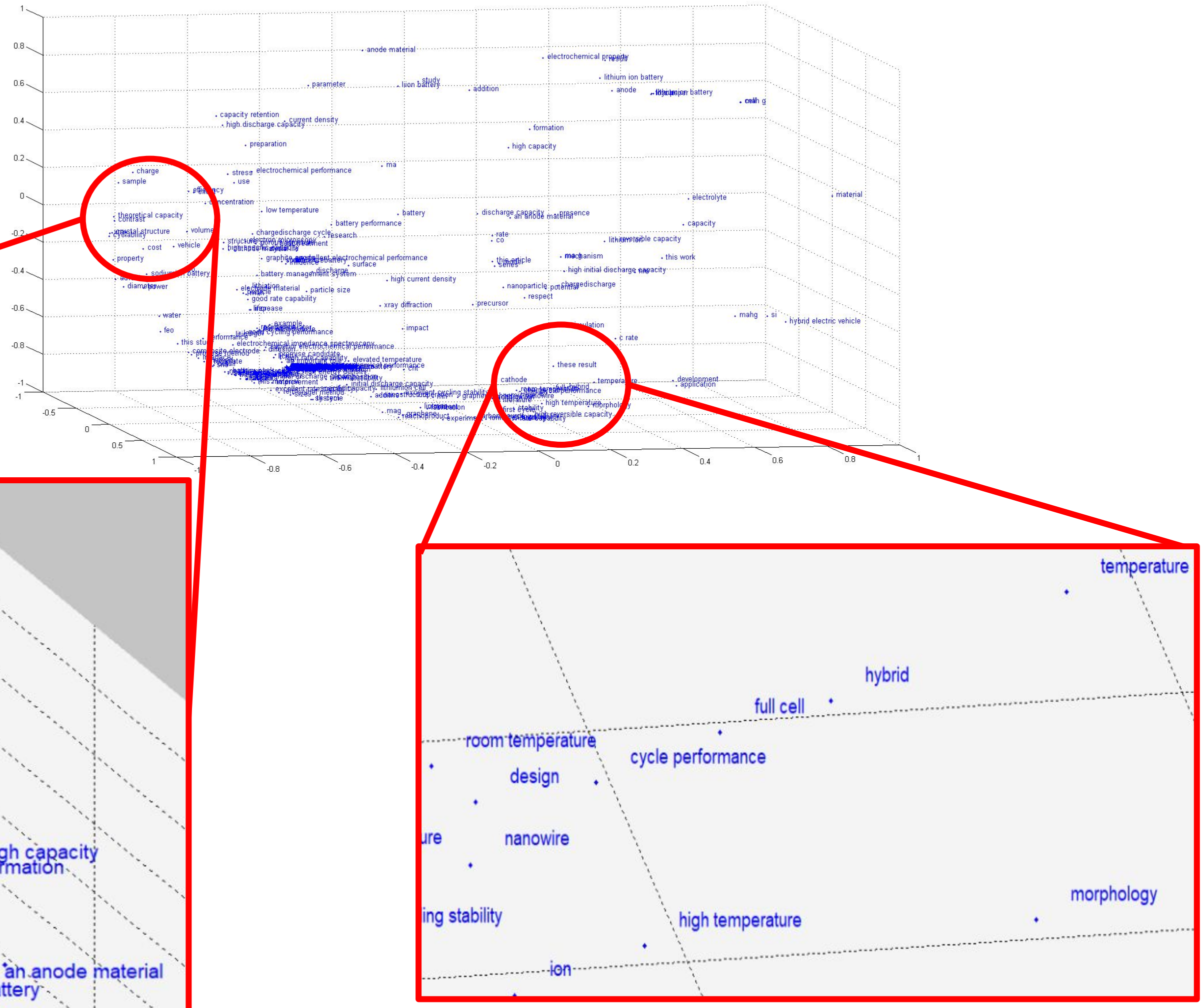
Architecture

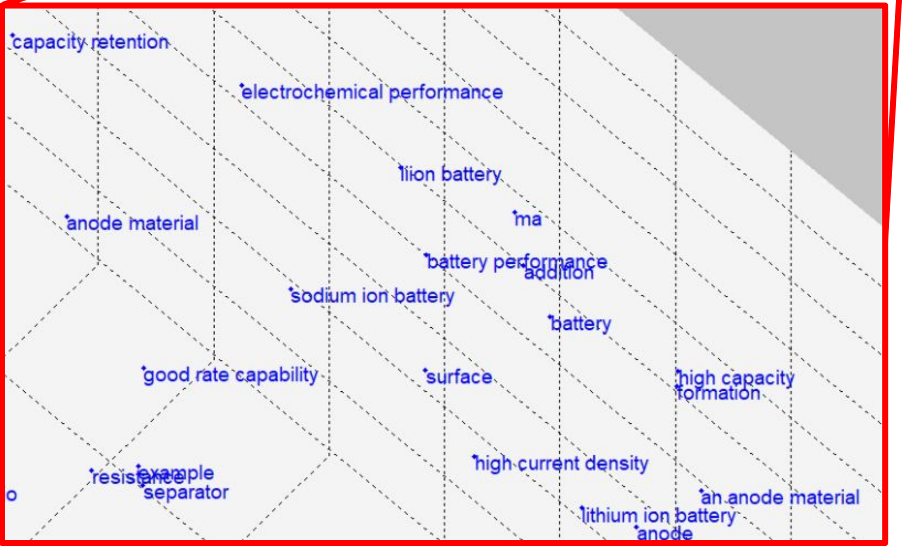
- Task 1: chunk -> embedding coordinates (== lookup table)
- Task 2: Approximate the function
  - (chunk 1 embedding, chunk 2 embedding) -> (chunk 3 embedding)
- Input layer is a product of context size and embedding dimension
  - Context size: 2, 3 words
  - Embedding dimension: 3-8
- Hidden layers: 1 or 2
- Output layer: embedding dimension

Training method

- Used both positive and negative samples
  - The ratio of their contribution to total loss is a hyperparameter
- Used TF-IDF weighting
  - More weight to a sample (only) if it shows up more frequently than its average frequency
- Restricted the embedding space to the unit sphere
  - Words can not get too far from each other

3D representation of the embedded expressions

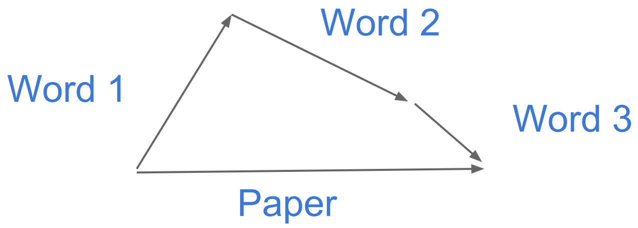




Step 5: Evaluate documents

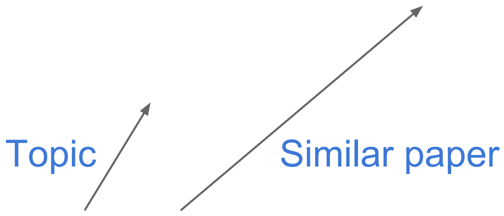
- Most simple model of an academic paper:

$$\text{Paper vector} = \sum \text{Word vectors}$$



- Text parts are similar if their vectors point in similar directions:

Give me the latest paper that talks about this topic:



Created a web-site

Lithium-ion battery with electrolyte additive

Show 5 most similar.

Show 5 most different.

Show Similar Papers

The selected paper was: Lithium-ion battery with electrolyte additive

The most similar papers were:

Lithium-ion battery with electrolyte additive Similarity: 0.99999999999999933387
Lithium ion secondary battery Similarity: 0.93236420941793840544
Lithium/air battery with variable volume insertion material Similarity: 0.8822923290115893824
Electrochemical instability of LiV 3 O 8 as an electrode material for aqueous rechargeable lithium batteries Similarity: 0.87508524535878073891
Electrochemical performance of high specific capacity of lithium-ion cell LiV 3 O 8//LiMn 2 O 4 with LiNO 3 aqueous solution electrolyte Similarity: 0.86019098656391679292

The least similar papers were:

Nanostructured anode materials for lithium ion batteries Similarity: -0.79009100344205551725
Engineering nanostructured electrodes away from equilibrium for lithium-ion batteries Similarity: -0.78682487841874115819
Fe 3 O 4@ porous carbon hybrid as the anode material for a lithium-ion battery: performance optimization by composition and microstructure tailoring Similarity: -0.78661168245106771746
Growth of MoS 2@ C nanobowls as a lithium-ion battery anode material Similarity: -0.78655207790604753093
MoO 2@ carbon hollow microspheres with tunable interiors and improved lithium-ion battery anode properties Similarity: -0.78526978336055108798

Step 5: Evaluate documents

- Similarity metric between papers based on the document vector was identified

⇨ Similar abstracts, papers can be queried from the system

- Example use case: “Find similar papers to my favourite paper”
  - Input: Fast sol–gel synthesis of LiFePO4/C for high power lithium-ion batteries for hybrid electric vehicle application (http://www.sciencedirect.com/science/article/pii/S0378775309010520)
  - Result 1: Synthesis of lithium-ion battery anode material Li4Ti5O12 by the microwave assisted sol-gel method (https://www.researchgate.net/publication/290300872_Synthesis_of_lithium-ion_battery_anode_material_Li4Ti5O12_by_the_microwave_assisted_sol-gel_method)
  - Result 2: Efficient microwave hydrothermal synthesis of nanocrystalline orthorhombic LiMnO2 cathodes for lithium batteries (www.sciencedirect.com/science/article/pii/S0013468610000587)

Finding relevant papers based on a patent

- The search space is limited to papers and the input is a patent.

- E.g: query patent:
  - Dynamically adaptive method for determining the state of charge of a battery
    https://www.google.com/patents/US7768233
    “The subject matter described herein generally relates to batteries used for electric or hybrid electric drivetrains, and more particularly relates to a method for recursively determining a state of charge in a battery system…”

- Most similar papers:

  - Development of toyota plug-in hybrid vehicle
  - (https://www.jstage.jst.go.jp/article/jaev/8/2/8_2_1399/_article)

  - Online estimation of lithium-ion battery remaining discharge capacity through differential voltage analysis
  - (http://www.sciencedirect.com/science/article/pii/S0378775314017510)

  - An adaptive sliding mode observer for lithium-ion battery state of charge and state of health estimation in electric vehicles
  - (http://www.sciencedirect.com/science/article/pii/S0967066116301149)

Conclusions

- Created a Proof-of-Concept for a tool that can be used for exploring the Lithium-ion battery patents and academic papers

- The tool can be trivially extended to other domains

- Codes are available upon request