Cross-Ontology Synonyms Detection

By

Ami Ladani (17CS60R85)

Under the supervision of

Dr. Pawan Goyal

Dept. of Computer Science and Engineering,

Indian Institute of Technology, Kharagpur

Nov 14, 2018

Outline

- Introduction
- Relevant Work
- Dataset and Preprocessing
- Methodologies
- Conclusion and Future Work
- References

Introduction

- One and the same thing can be described in many ways
- It is difficult to match online shoppers' unique and creative search queries to the products an e-commerce company sells
- Site selling ***sweatshirts***, should give desired results to the customers

  if they search for ***hoodies*** or ***sweats***, though ***sweatshirts*** and ***hoodies*** are not same, but these products are highly related
- Grammatical forms like ***T-shirt*** and ***T-shirts*** mean the same thing, just in different quantity
- Is it ***fitbit***, ***fit bit*** or ***fit-bit***?

Introduction

- **Objective:** Identifying synonyms using ontology mapping and semantic similarity across multiple ontologies, to solve the problem of ***zero search results*** and to ***enrich the glossary*** in a fashion domain

Relevant Work

- [Lin and Zhou, 2003; Yu and Wilbur, 2002] proposed to Identify synonyms among distributionally similar words
- [Van der Plas and Tiedemann, 2006] proposed to find synonyms using automatic word alignment and measures of distributional similarity
- [Saveski and Trajkovski, 2010] constructed valuable lexical resource WordNet
- [Gupta and Miller, 2015] proposed an unsupervised corpus-based conditional model Near-Synonym System (NeSS) for finding phrasal synonyms and near synonyms
- [Pak and Turemuratovich, 2015] proposed the Method of Synonyms Extraction from Unannotated Corpus using local and global contexts' statistics

DataSet and Preprocessing

- Training Dataset:
  - Unstructured Dataset:
    - Fashion E-commerce Websites' Catalog Data : Article Type, Gender, Brand, Retailer's Article Description, Style Name and Article Image
    - Size: 637,638 Entries
  - Structured Dataset:
    - Attribute-Value pairs for different Article Types
    - Size: 285 Entries

DataSet and Preprocessing

- Test Dataset:
  - Manually annotated set of synonyms for different Fashion Terms
  - Size: 273 Entries
- PreProcessing:
  - We apply tokenization, Part-Of-Speech tagging and Noun-phrase extraction
  - Our focus is to extract synonyms for fashion glossary terms which happen to be Noun phrases.
  - We only consider the vocabulary of words that occur at least 5 times in the corpus to ensure that the vectors have a minimum quality.

Word2Vec

- Using word2Vec embeddings to capture semantic relationships and different types of contexts in which words are used.
- If two words are used in same contexts then they may be synonyms or highly related to each-other

  eg. ***Fitness bands*** and ***Smart watches***

Word2Vec

- We experiment by taking whole article description corpus and running **CBOW** and **Skip-gram** models for different window size.
- We also try to learn embeddings by including and excluding Stop Words in the context window.
- We experiment by appending and not appending Article Type to the corresponding Article description.

Word2Vec : Results

Article Type	Ground Truth	Retrieved Synonyms
Dress Material	Dress material ,Dress Material	bottom fabric, Purple bottom fabric, unstitched dress material, Pink bottom fabric, Green bottom fabric, Orange bottom fabric, Black bottom fabric, printed kurta fabric, Red bottom fabric, kurta fabric
Capris	Caprijeans ,Capris	Pink capris, two capris, Blue capris, black knitted capris, Shorts, Track Pants, mid-rise lounge capris, mid-rise capris, capris, black mid-rise capris
Bracelet	Bracelets ,Bracelet ,Cuffs	Smooth band, Necklace, Bracelets, Icon Brand, Fine cuff, Bracelet, Bracelet pack, Anklet, Beaded chain, Icon Brand Pack

Word2Vec : Results

Window Size	Includes Article Name	Includes Stop Words	Avg Recall
1	No	No	0.055
1	No	Yes	0.051
10	No	No	0.066
Sentence Length	No	No	0.059
Sentence Length	**Yes**	**No**	**0.31**

FastText

- FastText is an extension to Word2Vec proposed by Facebook in 2016.
- Instead of feeding individual words into the Neural Network, FastText breaks words into several n-grams (sub-words).
- The word embedding vector for a word is the sum of all these n-grams.
- After training the Neural Network, we have word embeddings for all the n-grams given the training dataset.
- Rare words also can be properly represented since it is highly likely that some of their n-grams also appears in other words.

FastText

- We experiment by taking whole Articles' description corpus and running FastText for different configurations
- We experiment by taking window size 1, window size 10 and window size equal to full sentence length
- We experiment by appending and not appending article name to the corresponding article description
- We keep min n-gram length as 3 and max n-gram length as 6

FastText : Results

Article Type	Ground Truth	Retrieved Synonyms
Waistcoat	Waistcoats, Suit waistcoat, Waistcoat	adjustable cinch, Waistcoat by, waistcoat, Waistcoats, Tall waistcoat, Black waistcoat, waistcoat, Suit waistcoat, Waistcoat, Plus waistcoat
Face Moisturisers	Tinted Moisturisers, Face Moisturisers, Moisturisers	Moisturise, Moisturiser, Moisturisers, moisturisers, Face Wash and Cleanser, Moisturiser by, Moisturises, Fights, youthful-looking skin, skin
Casual Shoes	Desert Boots, Cowboy-/bikerboot, Pool shoes, Leather & Fashion Boots, Winter boots, Slip-ons, Shoes, Hiking shoes, Espadrille, Plimsolls, Going Out Shoes, Loafers, Boat Shoes	central lace-up, round-toed brown sneakers, central lace-ups Synthetic leather, central lace-up detail, central lace-ups Synthetic, round-toed blue sneakers, round-toed navy blue sneakers, round-toed black sneakers, round-toed pink sneakers, round-toed grey sneakers

FastText : Results

- Average recall for Top-30 similar articles using FastText method is more than double compared to best results using Word2Vec

Method	Avg Recall
Word2Vec	0.3
FastText	**0.7**

Dict2vec

- Both the previous methods implemented, suffer from a classic drawback of unsupervised learning: The lack of supervision between a word and those appearing in the associated contexts.
- It is likely that some terms of the context are not related to the considered word.
- On the other hand, the fact that two words do not appear together in any context of the training corpora is not a guarantee that these words are not semantically related
- Dict2vec adds new co-occurrences information based on the terms occurring in the definitions of a word. This latent word similarity and relatedness information introduces weak supervision that can be used to improve the embeddings.

Dict2vec

- If two words appear in the definitions of each-other, then they make a **strong pair**.
  eg. **Car** and **Vehicle**
- If only one word appear in the definition of other, then they make a **weak-pair.**
  eg. Car and Road
- Some weak pairs can be promoted as strong pairs if the two words are among the K closest neighbours of each other
- Positive Sampling: Dict2vec moves closer the vectors of words forming either a strong or a weak pair in addition to moving vectors of words co-occurring within the same window
- Controlled Negative Sampling: For each word in the vocabulary, it generate a set of k randomly selected unrelated words from the vocabulary, and separates their vectors.

Dict2vec

- Our corpus contains around 1531 unique words with more than 5 occurrences.
- Since there is no dictionary that contains a definition for all existing words, we combine several dictionaries(English version of Cambridge, Oxford, Collins, and dictionary.com) to get a definition for almost all of these words.
- Our approach focuses on noun phrases, so we consider only noun sense of all definitions for each word and concatenate results from all dictionaries, remove stop words and punctuation and lowercase all words.
- We train our model with the generated strong and weak pairs from these definitions.

Dict2vec: Results

- Average recall we found by using Word2vec embeddings is improved by Dict2vec
- The reason behind not so good performance of Dict2vec on our dataset is that formal dictionaries contained definitions for only few fashion terms and it generated 1 strong pair and 1578 weak pairs
- We believe that this results can further be improved if wikipedia dump is used as a dictionary. We will try this approach in future.

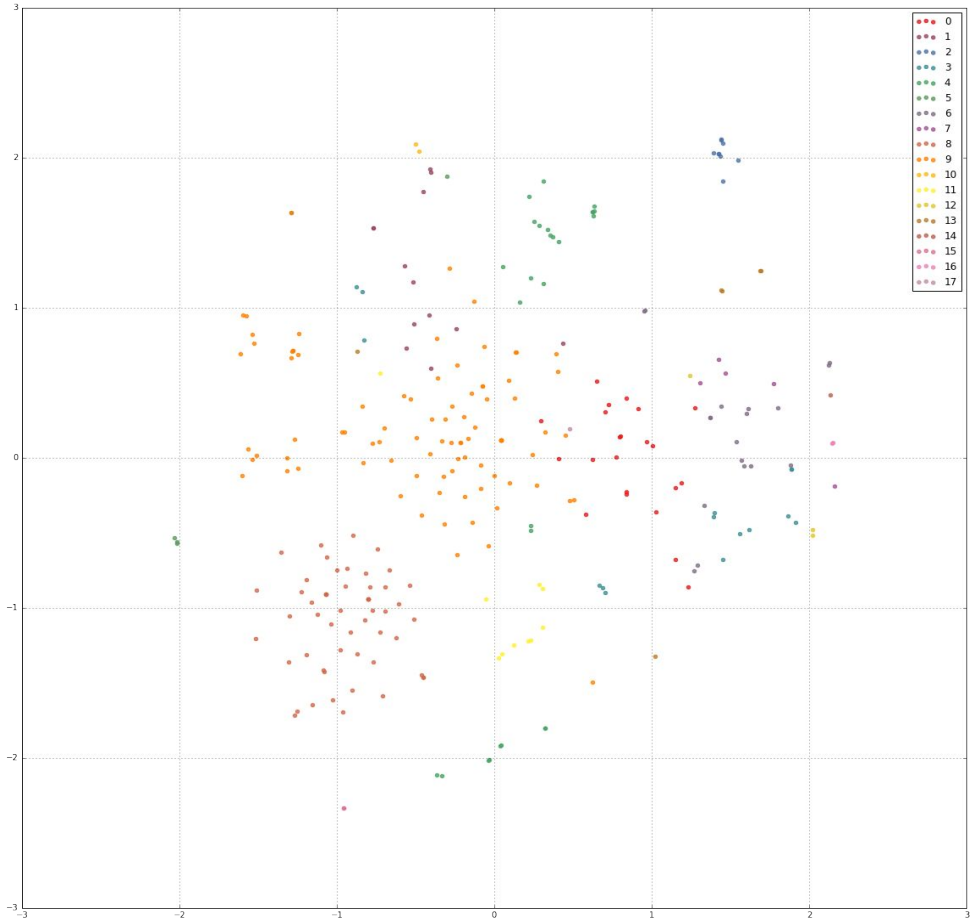
Method	Avg Recall
Word2vec	0.3
Dict2vec	**0.4**

K-means Clustering

- We have structured data for 285 different articles, each of them, contains attribute-value pairs.
- We have total 316 unique attributes among 285 different articles. When more than *x*-number of articles share same attribute we expand it to attribute-value pairs and then consider those pairs as features, otherwise we consider only attributes as features to avoid sparsity in feature vectors.
- For our dataset we get best results when value of *x* is 22.
- We experiment by setting different values for K, initializing cluster centers in a smart way to speed up convergence, changing the value of maximum number of iterations before convergence.

K-means Clustering : Results

Cluster Label	Cluster members
Cluster 2	churidar, Churidar and dupatta, harem pants, patiala, Patiala and Dupatta, pyjamas, Salwar, salwar and dupatta, shawl
Cluster 5	backpacks, duffel bag, rucksacks, trolley bag
Cluster 7	palazzos, shirts, shorts, trousers, tshirts
Cluster 12	dresses, skirts, tops
Cluster 15	fitness bands, smart watches



K-means Clustering : Results

- By evaluating the generated clusters manually it is found that articles sharing the same attribute-value features end up in same clusters.
- Articles which have more number of attributes, especially uncommon to most of other articles and common to few, are clustered correctly.
- The articles having very few attributes, end up in a random cluster having a maximum match.

Merging Word Vector Representations

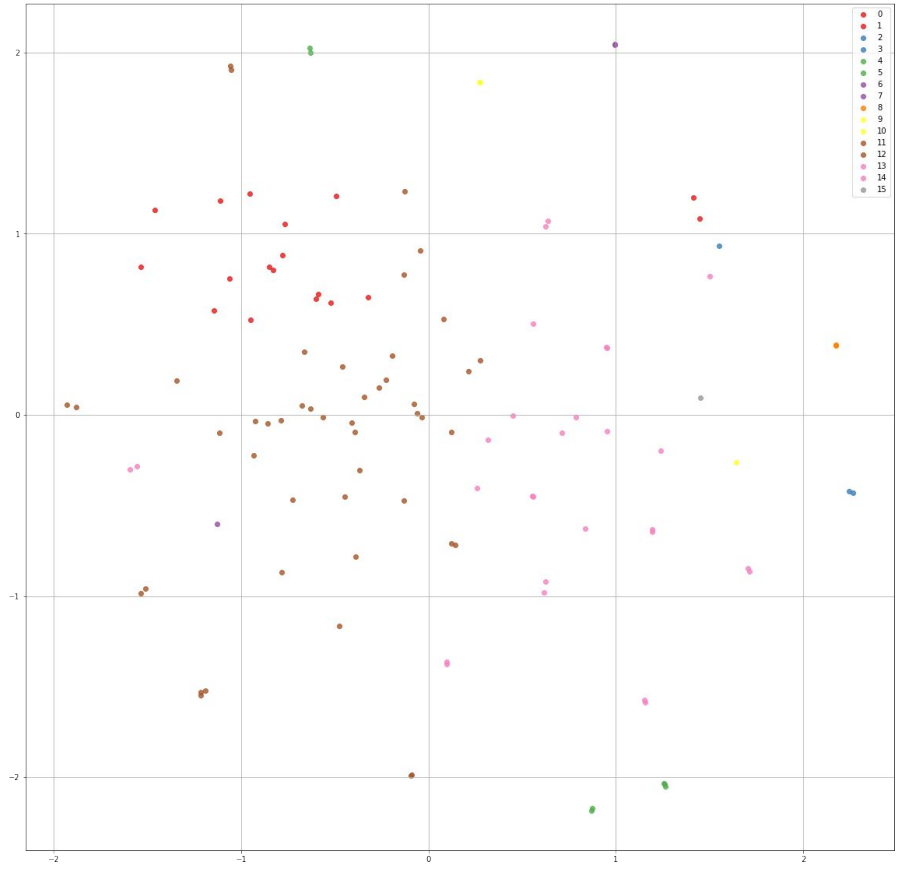
- Vector space word representations are learned from distributional information of words in large corpora.
- Although such statistics are semantically informative, they disregard the valuable information that is contained in structured data.
- We propose a method for refining vector space representations by merging them with multi-hot encodings learned using structured data(knowledge).

Merging Word Vector Representations

- We have word vectors for articles, learned using semantic information contained in our corpus as well as multi-hot representations learned using K-means clustering on structured data.
- Merge both the embeddings for each article and learn a new representation, containing semantic and structured information.
- It results in a very sparse and high dimensional feature vectors. We apply Principal component analysis on these feature vectors to reduce them to a small dimension that contains most of the information contained in original dimension.
- Apply K-means clustering on these feature vectors.

Merging Word Vector Representations: Results

Cluster Label	Cluster members
Cluster 1	Shawl, shawl, churidar, patiala, salwar
Cluster 3	skirts
Cluster 4	Tops, tops
Cluster 9	dresses
Cluster 11	trousers, Trousers



Merging Word Vector Representations: Results

- We find that merging reduces the clusters we found earlier using simple Kmeans, to fine grained clusters, by clustering only highly related terms with each other.
- This is a good result obtained by using semantic information and a small amount of structured data.

Results Summary

Method	Avg Recall
Word2vec	0.3
FastText	0.7
Dict2vec	0.4
K-means Clustering	0.49
Merging Embeddings	0.8

Conclusion

- We implemented several methods such as Word2Vec, FastText, Dict2vec, K-Means Clustering and merging word vectors.
- We find that FastText improves the results over Word2vec. Dict2vec seems to be promising in future using appropriate dictionary data.
- K-means clustering on structured data also perform better than Word2vec.
- Word vectors merged using structured data gives the best results.

Future Work

- Extracting structured knowledge from unstructured text and using it for a graph based embedding methods
- Retrofitting word embeddings using ConceptNet and other methods

References

[1] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword

information. CoRR.

[2] Gupta, Dishan, J. C. A. G. S. K. and Miller, D. (2015). Unsupervised phrasal near-synonym

gNinth AAAI Conference on Artificial Intelligence.

[3] Levy, O. and Goldberg, Y. (2014). Dependency-based word embeddings. ACL, pages 302—-308.

Lin, Dekang, S. Z. L. Q. and Zhou, M. (2003). Identifying synonyms among distributionally similar

words. IJCAI.

[4] Pak, Alexander Alexandrovich, S. S. N. A. S. Z. S. N. S. Z. E. K. and Turemuratovich, I. (2015). The

method of synonyms extraction from unannotated corpus. Digital Information, Networking,

and Wireless Communications (DINWC).

References

[5] Van der Plas, Lonneke, J. T. and Manguin, J. (2010). Automatic acquisition of synonyms for french using parallel corpora. Distributed Agent-based Retrieval Tools.

[6] Saveski, M. and Trajkovski, I. (2010). Automatic construction of wordnets by using machine translation and language modeling. 13th Multiconference Information Society, Ljubljana, Slovenia.

[7] T Mikolov, K Chen, G. C. and Dean, J. (2013a). Efficient estimation of word representations in vector space. Workshop at International Conference on Learning Representations(ICLR).

[8] T Mikolov, I Sutskever, K. C. G. C. and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems(NIPS).

[9] Tissier, J., Gravier, C., and Habrard, A. (2017). Dict2vec : Learning word embeddings using lexical dictionaries.

References

[10] Van der Plas, Lonneke, J. T. and Manguin, J. (2010). Automatic acquisition of synonyms for french using parallel corpora. Distributed Agent-based Retrieval Tools.

[11] Van der Plas, L. and Tiedemann, J. (2006). Finding synonyms using automatic word alignment and measures of distributional similarity. In Proceedings of the COLING/ACL.

[12] Yu, Hong, V. H. C. F. A. R. and Wilbur, W. J. (2002). Automatic extraction of gene and protein synonyms from medline and journal articles. AMIA Symposium.

[13] Plas, L. v. d. and Bouma, G. (2005). Syntactic contexts for finding semantically related words. LOT Occasional Series.

Q & A

Thank You!