An ontology-based approach to model qualitative world knowledge extracted from product reviews for use in QA systems

Thomas Huber

Monday 2nd July, 2018

Project Description

Extend the WordNetGraph [Silva et al.(2018)Silva, Freitas, and Handschuh] with information about features that certain classes of products have to improve the quality of the graph navigation algorithm and build a basic query system that uses textual entailment to find specific matching products for the query.
Which phones take the nicest pictures? $\rightarrow$ Relevant feature is *camera*

Problems

- Crawling sufficient amounts of product reviews takes a long time
- How to evaluate how / if extending the existing knowledge graph with product features is helpful
- Many reviews have spelling errors
- How to get from a user query to matching product features

Solutions

- Data: I used an existing dataset of 450212 reviews in the *cellphone* category from Amazon.com [1]
- Evaluation: BPI dataset, as used in [Silva et al.(2018)Silva, Freitas, and Handschuh], showing new and shorter paths that are created
- Spelling errors: Keep only words that appear in the frequency corpus of English Wikipedia [2]
- From query to features: Analyze example queries and reformulate them into a textual entailment problem

[1] `https://www.kaggle.com/PromptCloudHQ/amazon-reviews-unlocked-mobile-phones/data`, accessed 2018-06-20

[2] `http://wortschatz.uni-leipzig.de/de/download`, 1 million sentences

Feature extraction

Based on the work by [Scaffidi et al.(2007)Scaffidi, Bierhoff, Chang, Felker, Ng, and Jin]. Two major changes:

- I keep only nouns that appear in the frequency corpus (making the assumption they are correctly spelled if this is the case)
- I added a modified score that lowers the score of a feature **F** for product **P** if P has only few reviews

Graph navigation

Based on the work by [Silva et al.(2018)Silva, Freitas, and Handschuh].

- Start at node **S**
- Get all connected nodes **N**
- Calculate the semantic relatedness between the target **T** and **n** $\in$ N (using Indra [Sales et al.(2018)Sales, Souza, Barzegar, Davis, Freitas, and Handschuh])
- If the semantic relatedness is above a threshold, extract the *head words* from the definition of node **n** and add it to the path so far and add that new path to a stack
- A change I made here: If the relation is *supertype*, instead of getting the head words from the definition I use the node itself as the next node

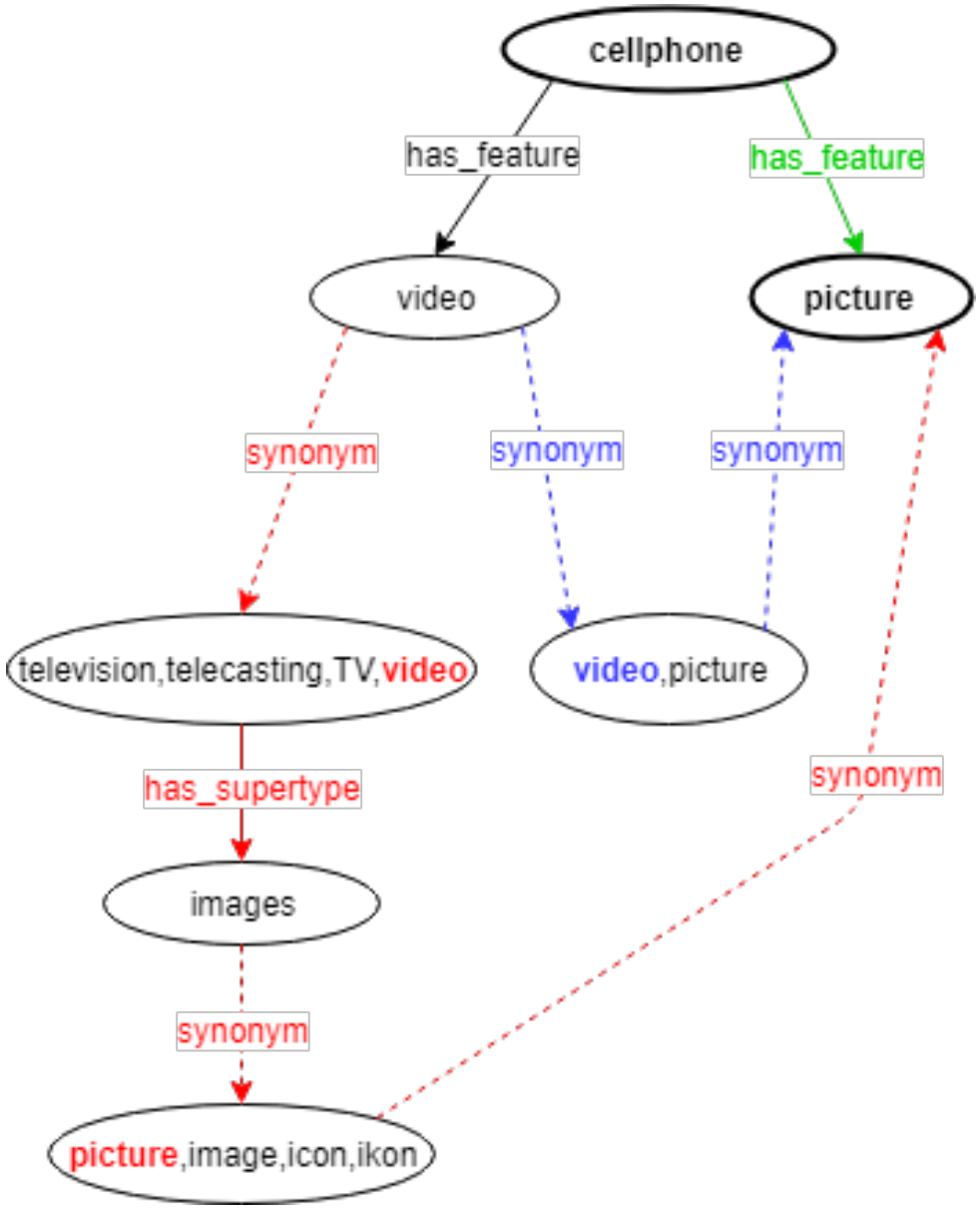

Figure: Example of three paths found between 'cellphone' and 'picture'

Evaluation

Table: Evaluation on BPI dataset

	Precision	Recall	F1
Original	0.65	0.54	0.59
My system	0.54	0.68	0.6

Table: Confusion matrix for BPI dataset

	Yes	No
Yes	67	58
No	31	93

Table: Evaluation BPI dataset without empty pairs

Precision	Recall	F1
0.71	0.68	0.7

Table: Confusion matrix, BPI dataset without empty pairs

	Yes	No
Yes	67	27
No	31	28

From query to textual entailment

I have formulated a few queries and used dependency parsing to analyze them. The query and the product features can be transformed into text and hypothesis pairs like so:

- Text: A *PRODUCT-CATEGORY* has feature *SOME-FEATURE*.
- Hypothesis: A *PRODUCT-CATEGORY* can *TARGET-VERB*.
- Hypothesis: A *PRODUCT-CATEGORY* has *TARGET-NOUN*.

Feature navigation

To find matching features, I look for paths between *PRODUCT-CATEGORY* and the target, which is either a verb or a noun as seen on the previous slide.
I then keep all paths that have *has feature* in them and show all products that have the feature and the scores the products have for those features.

Product recommendation example

Query: Which cellphones take the best pictures?

Minimum relatdness in graph navigation: 0.2

Maximum path length: 6

Maximum number of paths: 100

Query

picture | camera

('source', 'source', 'source', 'source', 'cellphone.n')
('cellphone.n has product feature (scaffidi) -> camera.n', 'cellular_telephone__cellular_phone__cellphone__cell__mobile_phone.n', 'has_feature', 'camera.n', 'camera.n')
('camera.n is contained in synset -> television_camera__tv_camera__camera.n', 'television_camera__tv_camera__camera.n', 'has_synonym', 'tv_camera.n', 'tv_camera.n'
('tv_camera.n is contained in synset -> television_camera__tv_camera__camera.n', 'television_camera__tv_camera__camera.n', 'has_synonym', 'television_camera.n', 'television_camera.n')
('television_camera.n is contained in synset -> television_camera__tv_camera__camera.n', 'equipment', 'has_diff_event', 'consisting of a lens system that focuses an im photosensitive mosaic that is scanned by an electron beam', 'mosaic.n')
('mosaic.n is contained in synset -> mosaic__arial_mosaic__photomosaic.n', 'arrangement', 'has_diff_qual', 'of aerial photographs forming a composite picture', 'pictu

Product	Score	Modified Score	Confidence
Otterbox Otterbox Defender Carrying Case for Samsung Galaxy S4 - Retail Packaging - Eden	5.0	5.0	1.0
Samsung Galaxy J7 SM- J700H/DS GSM Factory Unlocked Smartphone-Android 5.1- 5.5" AMOLED Display- International Version (White)	5.0	5.0	1.0

Summary

- Extended existing WordNet knowledge graph with features extracted from Amazon reviews
- This creates new paths for the graph navigation algorithm or shortens existing ones
- My system produces good results for entailment pairs it can check (precision of 0.71)
- Product recommendation component is very basic and can be extended
- One option to extend it is to explicitly create entailment pairs and return a list of features matching for a query

Bibliography I


Juliano Efson Sales, Leonardo Souza, Siamak Barzegar, Brian Davis, André Freitas, and Siegfried Handschuh.
Indra: A word embedding and semantic relatedness server.
In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).


Christopher Scaffidi, Kevin Bierhoff, Eric Chang, Mikhael Felker, Herman Ng, and Chun Jin.
Red opal: product-feature scoring from reviews.
In *Proceedings of the 8th ACM conference on Electronic commerce*, pages 182–191. ACM, 2007.

Bibliography II

Vivian S Silva, André Freitas, and Siegfried Handschuh. Recognizing and justifying text entailment through distributional navigation on definition graphs. 2018.