Set Expander: A Knowledge-based System for Entity Set Expansion

Weronika T. Adrian and Paweł Caryk

AGH University of Science and Technology, al. A. Mickiewicza 30, 30-059 Krakow, Poland

Keywords: Entity Set Expansion, Knowledge Graphs, Semantics, Knowledge-based Systems, Entity Recommendation.

Abstract: Entity Set Expansion (ESE) is a problem that underlies several important tasks, such as entity recommendation, query expansion, synonymy finding etc. Traditional strategies relied on corpus-based methods to recognize the intended category of the input words. But with the growing importance and visibility of knowledge graphs, new methods based on explicit knowledge representation have been put forward to solve the ESE problem. In this paper, we review the existing knowledge-based methods for entity set expansion and introduce a new online tool called Set Expander that uses semantic technologies and knowledge bases to solve the ESE problem efficiently. We present the algorithms and implemented techniques that ensure responsiveness and effectiveness of the tool. We analyze the strengths and weaknesses of the proposed solution and envision the future research directions.

1 INTRODUCTION

Entity Set Expansion (ESE) problem consists in finding entities, words or terms “similar” to the ones given as input. The problem implicitly assumes that one has to first understand what the input entities have in common, what they are, to what category they belong or what features they share; and then to retrieve, given some knowledge sources, more things “of the same kind” (Sarmento et al., 2007). The input terms are called the seeds, as we want to “grow” a bigger class, a set of entities from them. There are multiple domains of application of the ESE problem: from entity suggestion in recommendation engines, to user-guided dictionary creation (Kohita et al., 2020). “Traditional” approaches to entity set expansion relied on big textual corpus, within which the algorithms identified the patterns in which the seeds appeared, and matched those patterns with the corpus to find new entities. This kind of approach is called bootstrapping, because we start from a small set of seeds and iteratively learn new patterns and words that – by generating new patterns – guide the algorithms further. These algorithm were prone to so-called “concept drift” where mistakes at some point accumulated over time and several ways to mitigate these deviations were proposed. Apart from bootstrapping, several methods of learning features from text (e.g., news) corpus (Zhang and Liu, 2011), Web (Wang and Cohen, 2008; Hu and Jia, 2015) or social media media (Zhao et al., 2017) have been proposed.

With the renewed interest and availability of large structured knowledge bases, also new knowledge-based methods for ESE have been proposed. Instead of using textual patterns or co-occurrence statistics of terms in textual corpus, these new methods use structured knowledge bases to inform the category recognition and entity retrieval. In this paper, we review the newest approaches to Entity Set Expansion and present a new tool called Set Expander that communicates online with integrated knowledge graph BabelNet (Navigli and Ponzetto, 2012) and reason over the fetched knowledge.

2 A REVIEW OF KNOWLEDGE BASED APPROACHES TO ENTITY SET EXPANSION

2.1 Knowledge-based Methods for ESE

Knowledge graphs (Yan et al., 2018) are ontological resources that describe the universe of interest in terms of instances, classes and relationships among them. With the increasing interest in knowledge graphs, also new methods for ESE have been put forward. Zhang et al. (Zhang et al., 2017) proposed to use so-called semantic features which re-
2.2 Multifaceted Set Expansion

The input words may have multiple meanings and therefore several ways of expansion should be considered for a set of seeds. Rong et al. (Rong et al., 2016) addressed the problem of multifaceted set expansion, i.e., set expansion that takes into consideration different possible categories of expansion. The authors used skip-grams to learn “sibling” entities of the input terms and create so-called ego-networks. In these networks, clustering of sibling entities points to possible expansion directions. The algorithm fuse the networks with user-created ontologies, such as Wikipedia and word embeddings i.e., numerical vector representations of words.

Using online semantic resources was also proposed by Adrian et al. in (Adrian and Mann, 2018). The authors used logic programming approach to find the optimal ancestors of the input terms, based on network structure that contains information integrated from several knowledge bases. They used the semantic query language SPARQL (Pérez et al., 2009) to fetch the needed information, modelled the obtained knowledge as a graph and give it to the reasoner as an input of an optimization problem. A tool written in declarative ASP (Adrian et al., 2018) identifies, in the resulting network, clusters of meanings of the input words that serve as starting points for multifaceted set expansion.

Recently, the multi-faceted set expansion has also been addressed in (Zhu et al., 2019), where the authors propose a new framework called FUSE. The framework works in three phases, first identifying clusters that correspond to different facets, then optimizing the clusters by joining shared conceptualizations, and finally expanding the facets using a pre-trained models.

2.3 Recent Hybrid Proposals

The problem seems to enjoy a renewed interest in the recent years with multiply hybrid proposals. A joint model for learning Entity Set Expansion and attribute extraction has been proposed in (Zhang et al., 2016). A system based on lexical features and distributed representations has been proposed in (Yu et al., 2019). Several solutions using language probing (Zhang et al., 2020), pattern rank (Xiao et al., 2020) or enriching a corpus-based approach with so-called auxiliary sets and co-expansion (Huang et al., 2020) have been put forward. Finally, a bootstrapping system that relies on a neural network has been presented in (Yan et al., 2020).

2.4 Available Tools

Several years ago, the ESE functionality was available in the Google spreadsheet software. Unfortunately, this feature has been discontinued and the technology has ever since been protected by a patent. Thus, several attempts to put online an ESE tool have been undertaken. WordGrabBag.com is an online tool that let users enter a set of words and return a list of words “of the same kind”. It is based on the Word2Vec (Mikolov et al., 2013) method and neural language models to represent data from Wikipedia. The tool uses an optimized search algorithm to extract the new words relevant to the input query (more details can be found here: https://code.google.com/archive/p/word2vec/). SetExpan (Shen et al., 2017) is a corpus based tool that served as a base for a system for interactive dictionary generation (Kohita et al., 2020).

3 SET EXPANDER SYSTEM DESCRIPTION

3.1 General Assumptions

We assume an ontological knowledge base i.e., one that contains information about instances, classes and relationships. For clarity, we denote abstract concepts as “classes”, real world objects as “instances” and we will refer to any of them as “entities” (C ∪ I = E, where C is a set of concepts, I is a set of instances and E is a set of entities).

As for the ESE problem, we assume we get n words as input; each word may have m meanings (senses); each sense points to an instance in a knowledge base; each instance may (and usually do) belong to k0 (immediate) categories, that in turn are sub-classes of other k1 categories that are sub-classes of further k2 categories etc.

We divide the problem into two sub-problems:

---

1See http://wordgrabbag.com/.
1. to recognize the common category of the entities represented by the input words, and
2. to retrieve more relevant entities based on the identified category.

### 3.2 Technologies and Knowledge Resources

We have decided to use BabelNet (Navigli and Ponzetto, 2012) as a primary knowledge source. The BabelNet is a multilingual semantic encyclopedia, semi-automatically created from a number of established information resources such as Wikipedia, WordNet, Wikidata, OmegaWiki or GeoNames. Consequently, it integrates a paramount amount of knowledge about abstract concepts and real-world objects (so-called Named Entities), collecting their attributes and relationships with other entities.

BabelNet provided several APIs. After carefully considering all of them, we have decided to use the SPARQL interface to fetch the needed data. The language allows to formulate complex queries that we generate within the program.

### 3.3 Main Algorithms for ESE

To solve the first problem (see Sect. 3.1), we gather structured knowledge from our knowledge base and return a graph-based representation. We start with querying BabelNet with a SPARQL query built for the input words. For each word the query:
1. asks for all the possible senses (pointing to particular instances in BabelNet) for the given word, and
2. asks for the classes and their super-classes up to a required depth (from 1 to 5).

Using a single query we can fetch both the senses of the input words and the hierarchy of categories of the senses. On Listing 1, one can see an example query generated for a word “Java” and hierarchy depth 4:

```sql
SELECT DISTINCT ?a ?b WHERE {
  ?entries a lemon:LexicalEntry .
  ?entries lemon:sense ?sense .
  ?synset a skos:Concept .
  ?synset bn−lemon:synsetType ?synsetType .
  FILTER (?label="Java"@en || ?label="java"@en)
} .
```

Listing 1: Getting senses and categories from BabelNet.

The results of the query are organized into graphs (one for a sense of the input term) in which the nodes are the entities and the edges denote the semantic relations among them.

Then, we identify the most specific common category, taken into consideration all the combinations of the graphs created for each sense for all the words. As an example, consider two words w1 and w2. If we denote snsXY as a graph of categories for sense Y from word X, then from the following set of graphs:

\[
\begin{align*}
\text{sns11, sns12, sns13} & , \\
\text{sns21, sns22, sns23} & \\
\end{align*}
\]

we get the following combinations to analyze:

\[
\begin{align*}
\text{sns11, sns21} & , \\
\text{sns11, sns22} & , \\
\text{sns11, sns23} & \\
\text{sns12, sns21} & , \\
\text{sns12, sns22} & , \\
\text{sns12, sns23} & \\
\text{sns13, sns21} & , \\
\text{sns13, sns22} & , \\
\text{sns13, sns23} & \\
\end{align*}
\]

In graph theory, the problem is referred to as finding the lowest common ancestor and in Description Logics (Baader et al., 2004): finding the least common subsumer(s) for most specific concepts of the given instances. Our algorithm finds the common nodes in the combinations and return the one closest to the senses (see Fig. 1).

It may happen that there are more than one category equally distant from the senses. In this case, we keep all of these categories as possible users’ intents and expand all of them in the next step.

Then to retrieve more relevant entities, we perform the following query:

```sql
SELECT DISTINCT ?expand ?entry
  count(?related) as ?count
```
WHERE {
    ?expand skos:broader <http://babelnet.org/rdf/\{CategoryID\}>
    .
    ?expand bn--lemon: synsetType "NE" .
    ?expand skos:related ?related
    FILTER (starts(str(?entry), "http://dbpedia.org/resource/"))
}
GROUP BY ?expand ?entry
ORDER BY DESC(?count)
LIMIT 10

Listing 2: Getting relevant entities from BabelNet.

where:
- ?expand in query denotes ID of searching entities
- ?entry is link reference to entry in DB resources, from which name is extracted
- ?related denotes related ids to founded entity

The candidate entities must fulfill some constraints: they need to have the required category in “broader” relations and only “NE” (named entities) are desirable. To rank the candidate entities, for each expanded category only the 10 most related entities are returned.

Figure 1: Identifying the lowest common categories within combinations of graphs for (a) input words, that have (b) different senses that refer to instances in BabelNet and that belong to (c) classes that form a hypernymy hierarchy.

4 RESULTS AND DISCUSSION

4.1 Implementation Principles

We have implemented the SetExpander tool in Python, using Django framework. With respect to the first (unpublished) version of the tool, some improvements have been implemented:

- Calls to Babelnet API to retrieve the entities of a given categories are executed in a parallel loop, using Pool from multiprocessing library.
- In the prototype version of SetExpander querying of word synsets was divided into two parts: first, all senses of the given words were found, then for each sense, we searched for categories where it belongs. Now one query is performed for senses and their categories. Additionally, for each sense a graph of categories is built, with a depth from 1 to 5.
- The query to find the entities of the identified categories searches for ids and names. To improve the quality of the returned data and limit the response size a constraint to the query was added, the result are ordered by how many related relationships they have and then limited to 10.

The system code repository is available online.\(^2\)

The user interface is simple and consists of a search input field in which the user should enter the input words (see Fig. 2). Once the input words are entered

the tool calculates their common category or categories and return the results grouped by the category names (see Fig. 3).

4.2 Experiments and Conclusion

We have performed tests on the tool to check its behaviour, strengths and limits. Exemplary results for sets of various input words are presented in Table 1. In general, the response time of the application is satisfactory; for two words with four levels of graph depth, execution of find common categories takes approximately 10 milliseconds, for three words it takes approx. 100 ms and for four words –\(^2\)See https://anonymous.4open.science/r/e547777c-7d30-46c6-b988-36c82b3c05cc/.

\(^2\)See https://anonymous.4open.science/r/e547777c-7d30-46c6-b988-36c82b3c05cc/.
We discovered that increasing the depth of category hierarchy querying does not necessarily improve the quality of results. In fact, the quality depends strongly on the domain of the query and while for categories such as science (e.g., names of the planets) or geography (e.g., names of cities) it is sufficient to reach to depth 2, when it comes to people, even famous, we had to reach to depth 3 to get meaningful answers.

We could also observe that the richness of BabelNet comes with some noisiness and so for future work we plan to extend our research in two directions:

1. to analyze more relations, not only hypernymy (subclass-superclass relations) and to include them into the common category definition process, e.g., as in (Yang and Powers, 2005), where meronymy i.e., “part-of” relations are included into the definition of semantic similarity; and

<table>
<thead>
<tr>
<th>Seeds</th>
<th>Category</th>
<th>Expansion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>City</td>
<td>New york city, Frankfurt, Pretoria, Paris, Naples, Madrid, Berlin, Saint petersburg,</td>
</tr>
<tr>
<td></td>
<td>work</td>
<td>Alice’s adventures in wonderland, Don quixote, Asterix, Family guy, Alicia alono, Iliad, Tarzan, A christmas carol, Dracula</td>
</tr>
<tr>
<td></td>
<td>album</td>
<td>U-vox, Brilliant (album), 20 years of jethro tull, Systems of romance, Rage in eden, Visage (visage album), Metamatic, Blow up your video,</td>
</tr>
<tr>
<td></td>
<td>audio recording</td>
<td>Extended play, Hiroshima – rising from the abyss</td>
</tr>
<tr>
<td></td>
<td>film</td>
<td>Back to the future, The good, the bad and the ugly, Warner bros, Casino royale, New Hollywood, E.t. the extra-terrestrial, Titanic (1997 film), Kevin bacon, Jaws</td>
</tr>
<tr>
<td>Einstein, Newton, Galileo</td>
<td>human</td>
<td>John Paul II, Christ, Leonardo da vinci, Sigmund Freud, Gottfried Wilhelm Leibniz, Auguste Comte, American, Karl Marx, Albert Einstein, Mason</td>
</tr>
<tr>
<td></td>
<td>taxon</td>
<td>Epsilon 15</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>Lexus, Sara, Canute, American forces network, Eirik, Peter (given name), Robert the bruce, Gregory (given name), Adam (given name)</td>
</tr>
<tr>
<td>Neptune, Pluto, Saturn</td>
<td>planet</td>
<td>Pluto, Mars, Jupiter, Mercury, Venus, Earth, Saturn, Uranus, Neptune,</td>
</tr>
<tr>
<td></td>
<td>superior planet</td>
<td>Pluto, Mars, Jupiter, Uranus, Neptune</td>
</tr>
<tr>
<td></td>
<td>deity</td>
<td>Artemis, Dionysus, Aphrodite, Kore, Apollo, Hades, Devil, Poseidon, God, Athena</td>
</tr>
<tr>
<td></td>
<td>spiritual being</td>
<td>Devil</td>
</tr>
<tr>
<td></td>
<td>musical group</td>
<td>One direction, The byrds, Coldplay, S club 7, Ramones, Jls, Bullet for my valentine, Opeth, Nine inch nails, Underoath</td>
</tr>
<tr>
<td></td>
<td>music</td>
<td>Confederate states of america, Film score, Balkan music, Timeline of musical events, Wolfgang amadeus mozart, West side story, Emas, Karlheinz stockhausen, Last.fm</td>
</tr>
<tr>
<td></td>
<td>musical work</td>
<td>Nocturnes, op. 9 (chopin), Elijah (oratorio), In the hall of the mountain king, Appalachian spring, Symphonic dances (rachmaninoff), Masquerade (khachaturian), The art of fugue, Capriccio espagnol, Suite espa˜nola no. 1, op. 47</td>
</tr>
<tr>
<td></td>
<td>composition</td>
<td>Symphony no. 1 (mahler), Das lied von der erde, Symphony no. 4 (mahler), The tales of hoffmann, Symphony no. 2 (mahler), Symphony no. 3 (mahler), Book of lamentations, Paolo conte, Music of star wars, Symphony no. 8 (mahler)</td>
</tr>
<tr>
<td></td>
<td>album</td>
<td>Biophilia (album), Grammy award for album of the year, Post (bj¨ork album), Vespertine, The velvet rope, Debut (bj¨ork album), Extended play, Octavarium (album), Ray of light</td>
</tr>
<tr>
<td></td>
<td>individual</td>
<td>Krypto, God the son, Zapatista army of national liberation, Lapsed catholic</td>
</tr>
<tr>
<td></td>
<td>band</td>
<td>The byrds, Coldplay, Ramones, Underoath, Bullet for my valentine, Opeth, Nine inch nails, Billy talent, White lies</td>
</tr>
<tr>
<td></td>
<td>audio recording</td>
<td>Hiroshima – rising from the abyss, Extended play</td>
</tr>
</tbody>
</table>

around 1.1s. Implementing parallelism significantly improved the response time that mainly depends on communication with BabelNet.

WEBIST 2021 - 17th International Conference on Web Information Systems and Technologies
Figure 3: Results of the Set Expander tool for input: “Einstein, Newton, Bach”. As one could expect, the common category of this input is a general “last name”, as these people did not share a profession that would point to a more specific category. Surprisingly, there are also “craters” that are named after the entered names, and so more cats have been returned as a possible expansion.

2. to post-process the results of the expansion step to filter out some entities that are less common, have less “evidence” of their quality (currently, we have taken a step in this direction by ordering and selecting $k$ first candidate entities by the number of “related” relationships between the candidates and the seeds).

5 SUMMARY

In this paper, we reviewed the existing approaches to the ESE problem, with a special focus on the most recent ones. We have put forward a proposal of a new knowledge-based system called Set Expander that builds on semantic technologies and knowledge graph resources. The system is available online and upon configuration works in an interactive mode. The tool described in this paper is flexible and extendable and we plan to continue working on its improvements.

REFERENCES


