Common-Sense Knowledge for Natural Language Understanding: Experiments in Unsupervised and Supervised Settings

Luigi Di Caro(✉), Alice Ruggeri, Loredana Cupi, and Guido Boella

Department of Computer Science, University of Turin,
Corso Svizzera 185, Torino, Italy
{dicaro,ruggeri,cupi,boella}@di.unito.it

Abstract. Research in Computational Linguistics (CL) has been growing rapidly in recent years in terms of novel scientific challenges and commercial application opportunities. This is due to the fact that a very large part of the Web content is textual and written in many languages. A part from linguistic resources (e.g., WordNet), the research trend is moving towards the automatic extraction of semantic information from large corpora to support on-line understanding of textual data. An example of direct outcome is represented by common-sense semantic resources. The main example is ConceptNet, the final result of the Open Mind Common Sense project developed by MIT, which collected unstructured common-sense knowledge by asking people to contribute over the Web. In spite of being promising for its size and broad semantic coverage, few applications appeared in the literature so far, due to a number of issues such as inconsistency and sparseness. In this paper, we present the results of the application of this type of knowledge in two different (supervised and unsupervised) scenarios: the computation of semantic similarity (the keystone of most Computational Linguistics tasks), and the automatic identification of word meanings (Word Sense Induction) in simple syntactic structures.

1 Introduction

Recent Computational Linguistics advances are fully oriented towards the automatic extraction of semantic information through big and multilingual data analyses, since semantics help tasks such as disambiguation, summarization, entailment, question answering, and so forth. This explains the fortunate and growing area of semantic resources, often constructed with automatic approaches, when manual building of ontologies is not feasible on large scale. Semantic information extraction is currently approached by distributional analysis of linguistic items over specific contexts [1] or by starting from seeds and patterns to build ontologies from scratch [2]. In some cases, linguistic items are substituted by super-senses (i.e., top-level hypernyms) [3].

In recent years, the need and the opportunity of automatically extracting semantic information has been answered by Big Data. This led to the construction of very large semantic resources, such as ConceptNet [4], i.e., a semantic

M. Gavanelli et al. (Eds.): AI*IA 2015, LNAI 9336, pp. 233–245, 2015.
DOI: 10.1007/978-3-319-24309-2_18

graph that has been directly created from collection of unstructured common-sense knowledge asked to people contributing over the Web. In contrast with linguistic resources such as WordNet [5], ConceptNet contains semantic information that are more related to common-sense facts. For this reason, it has a wider spectrum of semantic relationships though a much more sparse coverage. For instance, among the more unusual types of relationships (24 in total), it contains semantic relations like “*ObstructedBy*” (i.e., referring to what would prevent it from happening), “ and *CausesDesire*” (i.e., what does it make you want to do). In addition, it also has classic relationships such as “*is_a*” and “*part_of*” as in other linguistic resources.

ConceptNet is a resource based on common-sense rather than linguistic knowledge and it contains much more function-based information (e.g., all the actions a concept can be associated with) contained in even complex syntactic structures. While it has been recognized as a very promising type of knowledge for many computational tasks, it is not significantly used yet due to its complexity. More in detail, ConceptNet has the following main problems:

1. *Specificity.* It contains very specific semantic information (e.g., $< knowledge - CapableOf - openhumanmind >$) that are difficult to integrate in automated tasks;
2. *Completeness.* It is not complete (due to the methodology used to build it), since semantic features are arbitrarily associated to only few of all the possible relevant concepts (e.g., ConceptNet contains $< jazz{-}IsA{-}styleofmusic >$) but not $< rock - IsA - styleofmusic >$);
3. *Correctness.* It contains pragmatics statements which are not semantically correct, e.g., $< cat - Antonym - dog >$;
4. *Relativity.* It has semantic features such as $< dog - HasProperty - small >$, which is not always true.

This paper presents an application of ConceptNet (as common-sense knowledge) in two different scenarios: the computation of similarity scores at word-level (one of the key task in Computational Linguistics) and the identification of the different meanings that can be associated with words in sentences. Although the tasks are of different type (supervised and unsupervised respectively), they are based on the same idea of replacing words with ConceptNet semantic information, to measure whether this knowledge has the potential to serve computational tasks even without any particular approach for data alignment, noise removal, semantic information propagation, etc.

2 Background and Related Work

Word Sense Disambiguation (WSD) [6] is maybe the most crucial Natural Language Processing task, since its aim is to capture the correct meaning to be associated to a word in a specific context. This allows to interpret the correct sense of a word, in order to understand how similar is with respect to other words. This permits a set of comparisions between texts, which is useful for many computational tasks such as Information Retrieval, Named Entity Recognition, Question Answering, and so forth. Generally speaking, systems are usually asked to compute similarity scores between pieces of texts at different granularity (word, sentence, discourse) [7].

In order to evaluate the similarity between texts, semantic resources are often used to consider a larger semantic basis to make more accurate comparisons. While linguistic resources such as WordNet and VerbNet constitute a highly-precise source of information, they often cover very few semantic relations usually focusing on taxonomic relations. Common-sense knowledge, on the other side, represents a much larger set of semantic features, which, however, is affected by noise and lack of completness.

If we consider the objects, agents and actions as *terms* in text sentences, we can try to extract their meaning and semantic constraints by using the idea of *affordances* [8]. The affordances of an object can be seen as the set of functionalities that it naturally communicates to the agents through its shape, size, and other phisical characteristics.

For instance, let us think to the sentence "*The squirrel climbs the tree*". In this case, we need to know what kind of subject "*squirrel*" is to figure out (and visually imagine) how the action will be performed. Let us now consider the sentence "*The elephant climbs the tree*". Even if there is no change in the grammatical structure of the sentence, the agent of the action creates some semantic problem. In fact, in order to climb a tree, the subject needs to fit to our mental model of "*something that can climb a tree*". In addition, this also depends on the mental model of "*tree*". Moreover, different agents can be both correct subjects of an action whilst they may produce different meanings in terms of how the action can be performed. A study of these language dynamics can be of help for many NLP tasks, e.g., Part-Of-Speech tagging as well as more complex operations such as dependency parsing and semantic relation extraction. Some of these concepts are latently studied in different disciplines related to statistics. Distributional Semantics (DS) [9] represents a class of statistical and linguistic analysis of text corpora that tries to estimate the validity of connections between subjects, verbs, and objects by means of statistical sources of significance.

3 A Large Common-sense Knowledge: ConceptNet

The Open Mind Common Sense[1] project developed by MIT collected unstructured common-sense knowledge by asking people to contribute over the Web. In this paper, we started focusing on ConceptNet [4], that is a semantic graph that has been directly created from it. In contrast with linguistic resources like WordNet [5], ConceptNet contains semantic information that are more related to common-sense facts.

[1] http://commons.media.mit.edu/

Table 1. Some of the existing relations in ConceptNet, with example sentences in English.

Relation	Example sentence
IsA	NP is a kind of NP.
LocatedNear	You are likely to find NP near NP.
UsedFor	NP is used for VP.
DefinedAs	NP is defined as NP.
HasA	NP has NP.
HasProperty	NP is AP.
CapableOf	NP can VP.
ReceivesAction	NP can be VP.
HasPrerequisite	NP—VP requires NP—VP.
MotivatedByGoal	You would VP because you want VP.
MadeOf	NP is made of NP.
...	...

For this reason, it has a wider spectrum of semantic relationships though a much more sparse coverage. For instance, among the more unusual types of relationships (24 in total), it contains information like "*ObstructedBy*" (i.e., referring to what would prevent it from happening), "and *CausesDesire*" (i.e., what does it make you want to do). In addition, it also has classic relationships like "*is_a*" and "*part_of*" as in most linguistic resources (see Table 1). ConceptNet is a resource based on common-sense rather than linguistic knowledge since it contains much more function-based information (e.g., all the actions a concept can be associated with) contained in even complex syntactic structures.

4 Supervised Experiment: Semantic Similarity

From a computational perspective, being words ambiguous, the disambiguation process (i.e., Word Sense Disambiguation) is one of the most studied tasks in Computational Linguistics. To make an example, the term *count* can mean many things like *nobleman* or *sum*. Using contextual information, it is often possible to make a choice. Again, this choice is done by means of comparisons among contexts, that are still made of words. In other terms, we may state that the *computational* part of almost all computational linguistics research is about the calculus of matching scores between linguistic items, i.e., *words similarity*.

4.1 Description of the Experiment

The experiment starts from the transformation of a word-word-score similarity dataset into a context-based dataset in which the words are replaced by sets of semantic information taken from ConceptNet. The aim is to understand if common-sense knowledge may represent a useful basis for capturing the similarity between words, and if it may outperform systems based on linguistic resources such as WordNet.

4.2 Data

We used the dataset SimLex-999 [10] that contains one thousand word pairs that were manually annotated with similarity scores. The inter-annotation agreement is 0.67 (Spearman correlation). We leveraged ConceptNet to retrieve the semantic information associated to the words of each pair, then keeping the simple intersection[2]. Then, we applied the same approach with the semantic information in WordNet. Note that we did not make any disambiguation of the words, so we used all the semantic data of all the possible senses of the words contained in the similarity dataset.

4.3 Running Example

Let us consider the pair *rice-bean*. ConceptNet returns the following set of semantic information for the term *rice*:

> [hasproperty-edible, isa-starch, memberof-oryza, atlocation-refrigerator, usedfor-survival, atlocation-atgrocerystore, isa-food, isa-domesticateplant, relatedto-grain, madeof-sake, isa-grain, isa-traditionally, receivesaction-eatfromdish, isa-often, receivesaction-cook, relatedto-kimchi, atlocation-pantry, atlocation-ricecrisp, relatedto-sidedish, atlocation-supermarket, receivesaction-stir, isa-staplefoodinorient, hasproperty-cookbeforeitbeeat, madeof-ricegrain, partof-cultivaterice, receivesaction-eat, derivedfrom-rice, isa-cereal, relatedto-white, hasproperty-white, hascontext-cook, relatedto-whitegrain, relatedto-food]

Then, the semantic information for the word *bean* are:

> [usedfor-fillbeanbagchair, atlocation-infield, atlocation-can, usedfor-nutrition, usedfor-cook, atlocation-atgrocerystore, memberof-leguminosae, usedfor-makefurniture, usedfor-grow, atlocation-foodstore, isa-legume, usedfor-count, isa-domesticateplant, hasproperty-easytogrow, partof-bean, atlocation-cookpot, isa-vegetableorperson-brain, atlocation-beansoup, atlocation-soup, atlocation-pantry, usedfor-plant, isa-vegetable, atlocation-container, usedfor-supplyprotein, atlocation-jar, usedfor-useasmarker, atlocation-field, derivedfrom-beanball, usedfor-shootfrompeashooter, atlocation-coffee, usedfor-fillbag, receivesaction-grindinthis, usedfor-beanandgarlicsauce, atlocation-beancan, usedfor-makebeanbag, usedfor-eat]

Finally, the intersection produces the following set (*semantic intersection*):

> [atlocation-atgrocerystore, isa-domesticateplant, atlocation-pantry]

Then, for each non-empty intersection, we created one instance of the type:

---

[2] In future works we will study more advanced types of matching.

<semantic intersection>, <similarity score>

and compute a standard *term-document* matrix, where the *term* is a semantic term within the set of semantic intersections and the *document* dimension represents the instances (i.e., the word pairs) of the original dataset. After this preprocessing phase, the *score* attribute is discretized into two bins:

– *non-similar* class - range in the dataset [0, 5]
– *similar* class - range in the dataset [5.1, 10]

4.4 Results

The splitting of the data into two clusters allowed us to experiment a classic supervised classification system, where a Machine Learning tool (a Support Vector Machine, in our case) has been used to learn a binary model for automatically classifying *similar* and *non-similar* word pairs. The result of the experiment is shown in Table 2. Noticeably, the classifier based on ConceptNet data has been able to reach a quite good accuracy (65.38% of correctly classified word pairs), considering that the inter-annotation agreement of the original data is only 0.67 (Spearman correlation). The use of WordNet produced a total F-measure of 0.582 (0.723 for the *non-similar* class and only 0.143 for the *similar* class). This demonstrates the potentiality of the semantic information contained in ConceptNet even if not structured and disambiguated as in WordNet.

Table 2. Classification results in terms of Precision, Recall, and F-measure with ConceptNet (CN) and WordNet (WN). With the use of WordNet, the system achieved 0.582 of total F-measure (0.723 for the non-similar class and only 0.143 for the similar class), while ConceptNet carried to higher accuracy levels.

Class	Prec. (CN)	Recall (CN)	F (CN)	Prec. (WN)	Recall (WN)	F (WN)
non-similar	0,697	0,475	0,565	0,574	0,978	0,723
similar	0,633	0,815	0,713	0,745	0,079	0,143
weighted total	**0,664**	**0,654**	**0,643**	0,649	0,582	0,467

Note that with ConceptNet, *similar* word pairs are generally easier to identify with respect to *non-similar* ones. On the other side, WordNet resulted to be not sufficient to generalize over *similar* word pairs.

5 Unsupervised Experiment: Word Sense Induction

In this section, we present an approach for automatic inducing senses (or meanings) related to the use of short linguistic constructions of the type *subject-verb-object*. As in the supervised experiment, we used the same approach, that is to replace words with the semantic information obtained from a semantic resource such as ConceptNet. The previous experiment demonstrated that the use of common-sense knowledge in this approach may outperform standard resources such as WordNet.

Instead of focusing on single words (as in the supevised setting), we experimented on a simple sentence-level task. The reason is twofold: 1) subject-verb-object are easily minable with very simple NLP parsers, and 2) they represent complete meanings (entire scenes, or actions). A *sense* is intended as in the well-known tasks named Word Sense Disambiguation (WSD) [6] and Word Sense Induction (WSI) [11], i.e., the meaning that a word assumes in a specific context. While WSD focuses on the classification of the meaning of a word among a given set of possible senses, WSI automatically finds senses as *clusters* of different contexts in which a word appears. Obviously, WSI systems are more complex to evaluate (as all clustering methods in general), eventhough [12] proposed a *pseudo-word* evaluation mechanism which is able to simulate the classic WSD scenario.

5.1 Description of the Experiment

The experiment is composed by three different phases: (1) the data preprocessing step with the generation of two transactional databases (transactions of items, as in the fields of Frequent Itemset Mining and Association Rules [13]) that we also call *semantic itemsets*; (2) the extraction of frequent, closed, and *diverse* itemsets (we will briefly introduce the meaning of all these names in the next paragraphs); and finally (3) the creation of *semantic verb models*, that generalize and automatically induce *senses* from entire linguistic constructions at sentence-level.

For a specific input verb v, we parse all the subject-verb-object (SVO) triples in the dataset[3] that have a higher frequency than a set threshold[4], taking into consideration morphological variations of v. Then, for each SVO triple, we replaced the subject-term and the object-term with the relative semantic features contained in ConceptNet. Table 3 shows an example of the information collected in this phase. Then, we associate each semantic information to a unique *id* and construct two transactional databases: one for the semantic information of the subjects, and one for the objects.

Once the transactional databases are built for a specific verb "v", we use techniques belonging to the field of Frequent Itemset Mining to extract *frequent patterns*, i.e, semantic features that frequently co-occur in our transactional databases[5].

This is done for both the transactional databases (subject and object databases associated to the verb 'v'). Since our aim is to capture all the linguistic *senses*, i.e., the different meanings connectable to the use of a specific

[3] More details about the used dataset are illustrated in a next section.

[4] In our experiments we considered SVO triples that occur at least 50 times in the whole ClueWeb09 corpus, in order to remove noisy data.

[5] In our experimentation, we used the library called SPMF for finding closed frequent itemsets[6], applying the CHARM algorithm [14].

Table 3. An example of subject- and object-terms semantic transformation for one triple of the verb "*to learn*" (*student-learns-math*). This represents one row of the two transactional databases.

Subject-term	Subject semantic features	Object-term	Object semantic features
student	*CapableOf*-study, *AtLocation*-at_school, *IsA*-person, *Desires*-learn, *PartOf*-class, *CapableOf*-read_book)	**math**	*IsA*-subject, *HasProperty*-useful_in_business, *UsedFor*-model_physical_world, ...

verb, we also need to obtain itemsets that cover all the items that are found in frequent itemsets. In other words, we want to extract *diverse itemsets*, i.e., a minimal set of frequent and closed itemsets that cover all the frequent items.

In the final phase, once obtained the frequent and diverse itemsets for both the two transactional databases, we connect all the subject-itemsets with all the object-itemsets, weighting the connection according to the their co-occurrences in the same triples of the original dataset.

The *semantic verb model* constructed for a specific verb "v" is thus a set of weighted connections between *frequent and diverse* semantic features belonging to the subjects of "v" and *frequent and diverse* semantic features of the objects of "v". On the one hand, this is a way to summarize the semantics suggested by the verb occurrences. On the other hand, it is also a result that can be used to generate new data by querying existing semantic resources with such semantic subject- and object-itemsets. Still, this can be done without looking for words similar to frequent subjects and objects, but by finding new subjects and objects that, even if not similar in general, have certain semantic information that fill the specific context.

The resulting models are automatically calculated, and they are very concise, since in all the large and sparse semantic space only few features are relevant to certain meanings (headed by the verb). This is also in line with what stated in [15] where the authors claimed that semantics is actually structured by low-dimensionality spaces that are covered up in high-dimensional standard vector spaces.

5.2 Data

We used a dataset of *subject-verb-object* (SVO) triples generated as part of the NELL project[7]. This dataset contains a set of 604 million triples extracted from

[7] http://rtw.ml.cmu.edu/resources/svo/

the entire dependency-parsed corpus *ClueWeb09* (about 230 billion tokens)[8]. The dataset also provides the frequency of each triple in the parsed corpus. The aim of the evaluation is two-folds: to demonstrate that the proposed approach is able to induce senses with a better efficacy than a classic word-based strategy, and to show that the resulting semantic model also drammatically reduces the dimensionality of the feature space, yet without mixing features.

5.3 Running Example

In order to better explain the whole data flow, we list some steps on an example verb (*to sing*). Within the dataset, we extracted 82 different triples (after morphologic and NER normalizations). The cardinality of subjects and objects is 95. The total number of semantic relations retrieved from ConceptNet is 8786. Then, with a minimum support of 0.05 (i.e., 5%), the output model of our approach is constituted by 4 diverse itemsets for the objects and 24 for the subjects, with an average itemset cardinality of 18.5 and 12.6 respectively, covering more than 50% of the semantic features of all the input triples. Thus, the size of the feature space for modeling the senses of the verb *to sing* goes from 95 words to 28 (4 + 24) semantic features (clusters of ConceptNet semantic information). In the next section we show the aggregate results on an extensive set of experiments using the *pseudo-word* evaluation proposed by [12].

The original dataset has several issues we needed to solve in order to garantee the significance of the evaluation. In detail, we removed very low-frequency triples containing mispelled words as well as meaningless sequences of characters that were not filtered out during the generation of the data. We manually tested different cutoffs to see the quality of the triples of the top-10 verbs in the corpus, selecting $m = 50$ as minimum frequency of the *subject-verb-object* triples in the *ClueWeb09* corpus.

Then, since the subject and object terms of the triples were not normalized, we merged all linguistic variations according to their morphological roots. Still, we integrated a Named Entity Recognition (NER) module to transform proper names into simple generic semantic classes, like people and organizations[9].

5.4 Results

Word Sense Induction systems are usually evaluated by using the pseudo-word strategy presented in [12]. The process proceeds as follows: first, two random words are selected, merged, and replaced by a unique token (for instance, the words *dog* and *apple* are replaced by the single pseudo-word *dogapple* in the corpus). Then, the WSI system is evaluated in terms of how well it separates the *dogapple*-instances into two correct clusters based on their context (as in a standard Word Sense Disambiguation scenario). In our case, since we use a large corpus of *subject-verb-object* instances, we evaluate our approach on verbs,

[8] http://lemurproject.org/clueweb09/

[9] We used the Stanford NLP toolkit available at http://www-nlp.stanford.edu/

Table 4. Average feature space cardinality for the three approaches (word-based *ww*, semantics-featured *sf*, semantic-model *sm*) on 100 random verb pairs within the top-100 frequent verbs.

F. Sp. Size (ww)	Min supp.	Sem. Loop	F. Sp. Size (sf)	F. Sp. Size (sm)
145,70	10%	1	4701,65	13,65
147,30	10%	2	2346,25	52,20
147,25	10%	3	1856,85	101,00
F. Sp. Size(ww)	Min supp.	Sem. Loop	F. Sp. Size (sf)	F. Sp. Size (sm)
146,90	20%	1	5159,70	4,55
145,10	20%	2	2232,40	14,10
143,50	20%	3	1863,80	41,75
F. Sp. Size (ww)	Min supp.	Sem. Loop	F. Sp. Size (sf)	F. Sp. Size (sm)
147,30	30%	1	4604,00	-
149,20	30%	2	2313,85	6,85
149,10	30%	3	1805,75	25,35

defining the context as their *subject-object* pairs. In particular, we extracted the top-100 frequent verbs in the dataset. From them, we randomly selected 100 verb pairs and replaced them with their pseudo-word. Some examples of pairs were *take-do*, *find-give*, *look-try*. On these evaluation data, we compared three approaches:

ww The *ww*-approach (*word-based*) constructs a dataset using subject and object terms as features.

sf The *sf*-approach (*semantic-featured*) constructs a dataset using the ConceptNet semantic information associated to subjects and objects as features, without mining frequent and diverse patterns from them.

sm The *sm*-approach (*semantic verb model*) constructs a dataset using the generated diverse itemsets as feature space.

For the *ww*-approach, for each verb pair under evaluation, we build a verb matrix $M^v_{I,J}$ where the rows represent the I-triples *subject-verb-object* containing the verb v while the columns $J = J_s \bigcup J_o$ are the union of subject and object terms (J_s and J_o respectively). For example, the triple *Robert-eats-vegetables* becomes an M^{eat}-row <*s-person*=1, *o-vegetable*=1> with only two non-zero features (notice that *Robert* has been substituted with the feature representing its associated Named Entity *person*, whereas *vegetables* has been linguistically-normalized to *vegetable*). The creation of the sf and the *sm* models is done in the same way. For the sf-approach we replaced subject and object terms with their ConceptNet semantic information, while for the *sm*-approach we replaced the features of the sf model with the generated diverse itemsets.

In this section we show and discuss the comparison of the three different models (*ww*, sf and *sm*) in terms of dimensionality reduction, classification accuracy and clustering adherence. We also experimented the usefulness of the

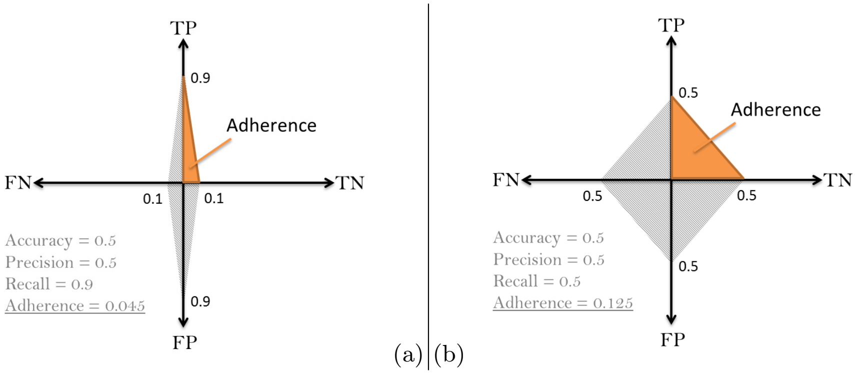

Fig. 1. An example sketch of the clustering evaluation results for the *ww*-approach (left figure (a)) and the *sf*- and *sm*-approach (right figure (b)). Accuracy and precision values are identical. Recall is superior for *ww*, in spite of the inability of the clustering process to separate the instances (almost all instances are in one single cluster). Clustering adherence permits to give light to the better intersection between clusters and classes obtained with our proposed approaches *sf* and *sm*.

recursive usage of ConceptNet. In detail, instead of simply replacing a word w with its semantic terms $< s_1, s_2, ..., s_n >$, we continue substituting each s_i with $< s_{i1}, s_{i2}, ..., s_{in} >$. We call this reiteration *semantic loop*.

Dimensionality Reduction. As shown in Table 4, the average size of the feature space is around 146 for the *ww*-approach, that is the average number of terms that fill the subject and object slots for the top-100 verbs in the collection (after normalization). When such terms are replaced by their ConceptNet semantic information (sf), the feature space becomes much larger. Actually, we only kept the top-1000 most common terms plus the terms that are as common as the least common term (i.e. ties are not broken). Note that while the semantic loop increases, the number of features for sf decreases. This is due to the fact that also ties on the 1000th term decrease with higher semantic loops, so the cutoff is able to operate earlier. Finally, the semantic model sm produces a very reduced feature space, compared to the ww and sf models.

Clustering. we used K-Means (with K=2) on the three models, then matching the resulting two clusters with the two actual *verb1* and *verb2* groupings. For each pair of instances i and j, it can happen one of the following 4 cases:

- i and j are clustered in the same cluster and they belong to the same class (true positive TP)
- i and j are clustered in the same cluster but they actually belong to different classes (false positive FP)
- i and j are clustered in different clusters but they actually belong to the same class (false negative FN)

Table 5. Average clustering adherence for the three approaches (word-based *ww*, semantics-featured *sf*, semantic-model *sm*) on 100 random verb pairs within the top-100 frequent verbs.

Clust. Adh. (ww)	Min supp.	Sem. Loop	Clust. Adh. (sf)	Clust. Adh. (sm)
7,90%	10%	1	13,57%	21,15%
7,46%	10%	2	15,69%	22,84%
5,71%	10%	3	15,95%	25,65%
Clust. Adh. (ww)	Min supp.	Sem. Loop	Clust. Adh. (sf)	Clust. Adh. (sm)
6,78%	20%	1	8,22%	22,38%
7,76%	20%	2	15,92%	24,48%
6,61%	20%	3	20,02%	26,10%
Clust. Adh. (ww)	Min supp.	Sem. Loop	Clust. Adh. (sf)	Clust. Adh. (sm)
8,46%	30%	1	11,40%	-
7,28%	30%	2	15,52%	22,46%
7,84%	30%	3	19,88%	25,35%

– i and j are clustered in different clusters and they belong to different classes (true negative TN)

The result of the clustering on the *ww*-approach is quite worthless in almost all the cases. In particular, the first cluster uses to contain one or two instances while the rest of the data remains in the second cluster. Thus, the computation of standard accuracy values carries to the same problem of the classification case (standard accuracy values are not meaningful). To overcome this inconvinient, we propose a novel measure, i.e., the *clustering adherence*. In words, its aim is to see how the feature space lets identify some coherent overlapping between the returned clusters and the actual classes in terms of TF and TN only (only in Information Retrieval systems FP and FN are actually important as TP and TN). Figure 1 shows the geometric representation of the measure. TP and TN form an *area of correctness*, that can be compared to its maximum value (when FP and FN are zero). Even if the three approaches have similar accuracy values, they present different clustering adherence. In particular, the clustering on the *ww*-model produces very low TP-TN areas with respect to sf and sm. Table 5 shows the complete results.

6 Conclusions and Future Works

Common-sense knowledge, differently from top-down semantic resources, has the potential to impact computational tasks thanks to the high number of semantic relations that is usually unfeasible to create manually. Nevertheless, the complexity in terms of specificity, completeness, correctness, and relativity makes it difficult to be used in numerous tasks. In this paper, we experimented two uses of common-sense knowledge in both an unsupervised and a supervised setting, by replacing words with the information associated to them in ConceptNet, a large common-sense semantic resource. The results highlights the power of this approach even before adequate management of the above-mentioned problematic common-sense type of semantic data. In future work, we will leverage techniques for data filtering and compression to improve quality in ConceptNet, evaluating the impact in main computational tasks.

References

1. Padó, S., Lapata, M.: Dependency-based construction of semantic space models. Computational Linguistics **33**(2), 161–199 (2007)
2. Navigli, R., Velardi, P., Faralli, S.: A graph-based algorithm for inducing lexical taxonomies from scratch. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence-Volume Volume Three, pp. 1872–1877. AAAI Press (2011)
3. Lenci, A.: Carving verb classes from corpora. In: Simone, R., Masini, F. (eds.) Word Classes, p. 7. John Benjamins, Amsterdam, Philadelphia (2010)
4. Speer, R., Havasi, C.: Representing general relational knowledge in ConceptNet 5. In: LREC, pp. 3679–3686 (2012)
5. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11), 39–41 (1995)
6. Stevenson, M., Wilks, Y.: Word-sense disambiguation. The Oxford Handbook of Comp. Linguistics, 249–265 (2003)
7. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing. MIT press (1999)
8. Gibson, J.J.: The Theory of Affordances. Lawrence Erlbaum (1977)
9. Baroni, M., Lenci, A.: Distributional memory: A general framework for corpus-based semantics. Computational Linguistics **36**(4), 673–721 (2010)
10. Hill, F., Reichart, R., Korhonen, A.: Simlex-999: Evaluating semantic models with (genuine) similarity estimation (2014). arXiv preprint arXiv:1408.3456
11. Denkowski, M.: A survey of techniques for unsupervised word sense induction. Language & Statistics II Literature Review (2009)
12. Schutze, H.: Dimensions of meaning. In: Supercomputing 1992., Proceedings, pp. 787–796. IEEE (1992)
13. Borgelt, C.: Frequent item set mining. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **2**(6), 437–456 (2012)
14. Zaki, M.J., Hsiao, C.J.: Charm: an efficient algorithm for closed itemset mining. In: SDM, Vol. 2, pp. 457–473. SIAM (2002)
15. Karlgren, J., Holst, A., Sahlgren, M.: Filaments of meaning in word space. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 531–538. Springer, Heidelberg (2008)