Common and Common-Sense Knowledge Integration for Concept-Level Sentiment Analysis

Erik Cambria
MIT Media Laboratory
cambria@media.mit.edu

Newton Howard
MIT Media Laboratory
nhmit@mit.edu

Abstract

In the era of Big Data, knowledge integration is key for tasks such as social media aggregation, opinion mining, and cyber-issue detection. The integration of different kinds of knowledge coming from multiple sources, however, is often a problematic issue as it either requires a lot of manual effort in defining aggregation rules or suffers from noise generated by automatic integration techniques. In this work, we propose a method based on conceptual primitives for efficiently integrating pieces of knowledge coming from different common and common-sense resources, which we test in the field of concept-level sentiment analysis.

Introduction

Concept-level sentiment analysis focuses on a semantic analysis of text through the use of web ontologies or semantic networks, which allow the aggregation of conceptual and affective information associated with natural language opinions. By relying on large semantic knowledge bases, such approaches step away from blind use of keywords and word co-occurrence count, but rather rely on the implicit features associated with natural language concepts. Unlike purely syntactical techniques, concept-based approaches are able to detect also sentiments that are expressed in a subtle manner, e.g., through the analysis of concepts that do not explicitly convey any emotion, but which are implicitly linked to other concepts that do so.

The bag-of-concepts model can represent semantics associated with natural language much better than bags-of-words. In the bag-of-words model, in fact, a concept such as `cloud computing` would be split into two separate words, disrupting the semantics of the input sentence (in which, for example, the word `cloud` could wrongly activate concepts related to `weather`). The analysis at concept-level allows for the inference of semantic and affective information associated with natural language opinions and, hence, enables a comparative fine-grained feature-based sentiment analysis. Rather than gathering isolated opinions about a whole item (e.g., iPhone 5s), users are generally more interested in comparing different products according to their specific features (e.g., iPhone 5s' vs Galaxy S5's touchscreen), or even sub-features (e.g., fragility of iPhone 5s' vs Galaxy S5's touchscreen). In this context, the construction of comprehensive common and common-sense knowledge bases is key for feature-spotting and polarity detection, respectively. Common-sense, in particular, is necessary to properly deconstruct natural language text into sentiments for example, to appraise the concept `small room` as negative for a hotel review and `small queue` as positive in a patient opinion, or the concept `go read the book` as positive for a book review but negative for a movie review.

Collecting and aggregating such kind of multi-source and multi-domain knowledge, however, is a formidable task as it requires advanced natural language processing technologies such as information extraction, word-sense disambiguation, and analogical reasoning. In this work, we propose a technique for integrating common and common-sense sources, which leverages on conceptual primitives to efficiently aggregate pieces of knowledge and, hence, we employ it in the context of concept-level sentiment analysis.

The rest of the paper is organized as follows: next section presents the different kinds of knowledge and knowledge sources that need to be aggregated; the following section describes in detail the proposed integration technique; the third section proposes an evaluation in the context of concept-level sentiment analysis; lastly, a final section concludes the paper and suggests directions for future work.

Knowledge Sources

In standard human-to-human communication, people usually refer to existing facts and circumstances and build new useful, funny, or interesting information on the top of those. This common knowledge includes information usually found in news, articles, debates, lectures, etc. (factual knowledge), but also principles and definitions that can be found in collective intelligence projects such as Wikipedia (vocabulary knowledge).

Moreover, when people communicate with each other, they rely on similar background knowledge, e.g., the way objects relate to each other in the world, people's goals in their daily lives, and the emotional content of events or situations. This taken-for-granted information is what is termed common-sense – obvious things people normally know and usually leave unstated (Cambria et al. 2009).

Copyright © 2014, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

The difference between common and common-sense knowledge can be expressed as the difference between knowing the name of an object and understanding the same object's purpose. For example, you can know the name of all the different kinds or brands of 'pipe', but not its purpose nor the method of usage. In other words, a 'pipe' is not a pipe unless it can be used (Cambria and White 2014).

Common Knowledge Sources

Attempts to build a common knowledge base are countless and include both resources crafted by human experts or community efforts, such as DBPedia (Bizer et al. 2009), a collection of 2.6 million entities extracted from Wikipedia, and Freebase (Bollacker et al. 2008), a social database of 1,450 concepts, and automatically-built knowledge bases, such as YAGO (Suchanek, Kasneci, and Weikum 2007), a semantic knowledge base of 149,162 instances derived from Wikipedia Infoboxes and WordNet, NELL (Carlson et al. 2010), with 242,000 beliefs mined from the Web, and Probase (Wu et al. 2012), Microsoft's probabilistic taxonomy counting about 12 million concepts learned iteratively from 1.68 billion web pages in Bing web repository.

Common-Sense Knowledge Sources

One of the biggest projects aiming to build a comprehensive common-sense knowledge base is Cyc (Lenat and Guha 1989). Cyc, however, requires the involvement of experts working on a specific programming language, which makes knowledge engineering labor-intensive and time-consuming. A more recent and scalable project is Open Mind Common Sense (OMCS), which is collecting pieces of knowledge from volunteers on the Internet by enabling them to enter common-sense into the system with no special training or knowledge of computer science. OMCS exploits these pieces of common-sense knowledge to automatically build ConceptNet (Speer and Havasi 2012), a semantic network of 173,398 nodes. Other projects that fall under this umbrella include WordNet, with its 25,000 synsets, and derivative resources such as WordNet-Affect.

Knowledge Integration

The aggregation of common and common-sense knowledge bases is designed as a 2-stage process in which different pieces of knowledge are first translated into RDF triples and then inserted into a matrix through the use of conceptual primitives. In particular, we use Shank's dependency primitives (Shank and Tesler 1969) plus SUBSUME, ARCHETYPE and AROUSE, three primitives that we need for categorical, analogical and affective reasoning, respectively.

We use Shank's primitives to generalize most actions usually performed by humans, animals, or agents. SUBSUME includes subsumption relationships such as *IsA*, *KindOf*, *HasCategory*, *DefinedAs*, and *SymbolOf*. ARCHETYPE represents archetypal relationships such as *HasA*, *HasProperty*, *PartOf*, *MadeOf*, and *CapableOf*. AROUSE includes affective relationships such as *MakesFeel*, *HasEmotion*, and *GeneratesMood*. The full list of conceptual primitives is as follows:

1. **SUBSUME**: *The notion of belonging to a category*
2. **ARCHETYPE**: *The notion of having canonical features*
3. **AROUSE**: *The act of making someone feel an emotion*
4. **PTRANS**: *The transfer of location of an object*
5. **ATRANS**: *The transfer of ownership or control*
6. **MTRANS**: *The transfer of mental information*
7. **MBUILD**: *The construction of new information*
8. **ATTEND**: *The act of focusing attention toward an object*
9. **GRASP**: *The grasping of an object*
10. **PROPEL**: *The application of a physical force to an object*
11. **MOVE**: *The movement of a body or bodypart*
12. **INGEST**: *The taking in of an object*
13. **EXPEL**: *The expulsion of an object*
14. **SPEAK**: *The act of producing sound*

All such primitives are used to define basic relations between natural language concepts that we exploit to generalize more complex and opaque conceptual relationships and, hence, obtain a more compact matrix representation of common and common-sense knowledge (Table1).

For example, a piece of knowledge such as "René Magritte is an artist" is first translated into the RDF triple <*René Magritte* - IsA - *artist*> through linguistic patterns. Its confidence score is then inserted in the matrix SUBSUME at the row named *René Magritte* and column named *artist*. The confidence score is either calculated according to how many times the sentence was found in a corpus, as in the case of Probase, or it was assessed as valid by annotators, as in the case of ConcetpNet. Similarly, a piece of knowledge such as "Horses have tails" becomes <*horse* - HasA - *tail*> and, hence, is inserted in the matrix ARCHETYPE as [*tail*; *horse*; confidence score].

The purpose of such an integration process is two-fold: firstly, it provides a shared representation for common and common-sense knowledge to be efficiently stored and, hence, used for reasoning; secondly, it performs 'conceptual generalization' of common relation types. Such generalization enables the representation of pieces of knowledge in a common framework, which allows the fusing of data from different sources without requiring ontology alignment and to combine data arising from multiple knowledge bases during reasoning (Kuo and Hsu 2012).

Table 1: The SUBSUME matrix enables the semantic clustering of concepts sharing the same subsumbtion features

SUBSUME	*artist*	*vehicle*	*phone*	*man*	*car*	...
René Magritte	0.98	0	0	0.77	0	...
iPhone 5s	0	0	0.93	0	0	...
Leonardo	0.9	0	0	0.83	0	...
Harley-Davidson	0	0.91	0	0	0	...
Galaxy S5	0	0	0.8	0	0	...
Cadillac	0	0.87	0	0	0.7	...
Dune Buggy	0	0.79	0	0	0.6	...
...	...	...	...	...	...	...

Conceptual generalization, moreover, allows the sparseness of common and common-sense knowledge matrices to be consistently reduced. The SUBSUME matrix, for example, aggregates pieces of knowledge concerning *IsA*, *KindOf*, *HasCategory*, and *SymbolOf* relationships that, if represented as a matrix singularly, would be too sparse to be processed and to enable efficient analogical reasoning.

Evaluation

In order to evaluate our integration technique, we reduce the dimensionality of the SUBSUME matrix (for even better generalization) and gauge its capacity for analogical reasoning on a Twitter dataset by comparing it with the performance of singular subsumption resources.

Analogical Reasoning

In order to more-compactly represent the information contained in the SUBSUME matrix $S \in \mathcal{R}^{m \times n}$ and encode the latent semantics between its instances, a multi-dimensional vector space representation is built by applying truncated singular value decomposition (SVD). The resulting lower-dimensional space represents the best approximation of S, in fact:

$$\begin{aligned} \min_{\tilde{S}|rank(\tilde{S})=k} |S - \tilde{S}| &= \min_{\tilde{S}|rank(\tilde{S})=k} |\Sigma - U_k^T \tilde{S} V_k| \\ &= \min_{\tilde{S}|rank(\tilde{S})=k} |\Sigma - D_k| \end{aligned}$$

where S has the form $S = U\Sigma V^T$, $\tilde{S}$ has the form $\tilde{S} = U_k D_k V_k^T$ ($U_k \in \mathcal{R}^{m \times k}$, $V_k \in \mathcal{R}^{n \times k}$, and $D_k \in \mathcal{R}^{k \times k}$ is diagonal matrix), and k is the lower dimension of the latent semantic space. From the rank constraint, i.e., D_k has k nonzero diagonal entries, the minimum of the above statement is obtained as follows:

$$\min_{\tilde{S}|rank(\tilde{S})=k} |\Sigma - D_k| = \min_{d_i} \sqrt{\sum_{i=1}^{n} (\sigma_i - d_i)^2} =$$

$$= \min_{d_i} \sqrt{\sum_{i=1}^{k} (\sigma_i - d_i)^2 + \sum_{i=k+1}^{n} \sigma_i^2} = \sqrt{\sum_{i=k+1}^{n} \sigma_i^2}$$

Therefore, $\tilde{S}$ of rank k is the best approximation of S in the Frobenius-norm sense when $\sigma_i = d_i$ $(i = 1, ..., k)$ and the corresponding singular vectors are the same as those of S. If all but the first d principal components are discarded and $\tilde{S}_U = U_k S_k$ is considered, a space in which common and common-sense instances are represented by vectors of k coordinates is obtained. These coordinates can be seen as describing instances in terms of 'eigenconcepts' that form the axes of the vector space, i.e., its basis $e = (e^{(1)}, ... e^{(k)})^T$.

A trial-and-error approach is used to find that the best compromise is achieved when k assumes values around 500. Such a 500-dimensional vector space can be used for making analogies (given a specific instance, find the instances most semantically related to it), for making comparisons (given two instances, infer their degree of semantic relatedness), and for classification purposes (given a specific instance, assign it to a predefined cluster).

Semantic Clustering

In order to cluster different SUBSUME instances according to their semantic relatedness, a sentic medoids approach (Cambria et al. 2011) is employed. Unlike the k-means algorithm (which does not pose constraints on centroids), sentic medoids do assume that centroids must coincide with k observed points, which allows to better cluster a vector space of common-sense knowledge. The sentic medoids approach is similar to the partitioning around medoids (PAM) algorithm, which determines a medoid for each cluster selecting the most centrally located centroid within that cluster. Unlike other PAM techniques, however, the sentic medoids algorithm runs similarly to k-means and, hence, requires a significantly reduced computational time.

Generally, the initialization of clusters for clustering algorithms is a problematic task as the process often risks to get stuck into local optimum points, depending on the initial choice of centroids. For this study, however, the most representative instances of SUBSUME (i.e., the matrix columns having higher number of entries) are used as initial centroids. For this reason, what is usually seen as a limitation of the algorithm can be seen as advantage for this study, since what is being sought is not the k centroids leading to the best k clusters, but indeed the k centroids identifying the top k hub concepts (i.e., the centroids should not be 'too far' from the most representative instances of these concepts).

Therefore, given that the distance between two points in the space is defined as $D(e_i, e_j) = \sqrt{\sum_{s=1}^{d'} \left(e_i^{(s)} - e_j^{(s)}\right)^2}$, the adopted algorithm can be summarized as follows:

1. Each centroid $\bar{e}_i \in \mathbb{R}^{d'}$ $(i = 1, 2, ..., k)$ is set as one of the k most representative instances of the top hub concepts;
2. Assign each instance e_j to a cluster $\bar{e}_i$ if $D(e_j, \bar{e}_i) \leq D(e_j, \bar{e}_{i'})$ where $i(i') = 1, 2, ..., k$;
3. Find a new centroid $\bar{e}_i$ for each cluster c so that $\sum_{j \in Cluster\ c} D(e_j, \bar{e}_i) \leq \sum_{j \in Cluster\ c} D(e_j, \bar{e}_{i'})$;
4. Repeat step 2 and 3 until no changes on centroids are observed.

Twitter Classification

In order to assess the accuracy of the proposed integration technique, we developed an opinion-mining engine that detects opinion targets in tweets and associates a polarity value to each of these. Such an engine consists of four main components: a pre-processing module, which performs a first skim of tweets; a semantic parser, whose aim is to extract concepts from text; a target spotting module, which identifies opinion targets; and a polarity detector, which determines if each tweet is positive or negative.

The pre-processing module firstly interprets special punctuation, complete upper-case words, cross-linguistic onomatopoeias, exclamation words, negations, degree adverbs, and emoticons. Secondly, it converts text to lower-case and, after lemmatizing it, splits the opinion into single clauses according to grammatical conjunctions and punctuation.

Then, the semantic parser deconstructs text into small bags of concepts (SBoCs) using a lexicon based on sequences of lexemes that represent multiple-word concepts extracted from the knowledge base selected at each run (SUBSUME, *IsA*, *KindOf*, and *HasCategory*, respectively). These n-grams are not used blindly as fixed word patterns but exploited as reference for the module, in order to extract concepts like `buy christmas present` from information-rich sentences such as "I bought a lot of very nice Christmas presents".

The target spotting module individuates one or more opinion targets, such as *people*, *places*, and *events*, from the input concepts. This is done by projecting the concepts of each SBoC into $\tilde{S}_U$, clustered according to SUBSUME's hub concepts. The categorization does not consist in simply labeling each concept, but also in assigning a confidence score to each category label, which is directly proportional to the value of belonging (dot product) to a specific conceptual cluster.

The polarity detector, finally, exploits the Sentic API (Cambria et al. 2014) to determine if a tweet is positive or negative by averaging polarity scores associated with each extracted concept.

The evaluation resource is a collection of 3,000 tweets crawled from Bing web repository by exploiting Twitter hashtags as category labels. In particular, hashtags about *electronics* (e.g., iPhone, XBox, Android, and Wii), *companies* (e.g., Apple, Microsoft, and Google), *countries*, *cities*, *operative systems*, and *cars* are selected. A comparative evaluation was performed against the richest subsumption resources, i.e., *IsA*, *KindOf*, and *HasCategory*. Results are reported in Table 2.

Conclusion

The integration of multi-source and multi-domain knowledge is often a problematic issue as it either requires a lot of manual effort in defining aggregation rules or suffers from noise generated by automatic integration techniques.

In this work, we proposed an integration technique based on conceptual primitives for efficiently integrating pieces of knowledge coming from different common and commonsense resources. The 'conceptual generalization' introduced by the proposed method not only allows the fusing of data from different sources without requiring ontology alignment, but also allows the sparseness of common and common-sense knowledge matrices to be consistently reduced.

Table 2: Target-dependent polarity detection accuracy of different subsumption resources measured on a Twitter dataset

	HasCategory	*KindOf*	*IsA*	SUBSUME
electronics	33.7%	44.1%	78.1%	79.7%
companies	27.5%	53.3%	80.0%	83.4%
countries	39.9%	67.6%	86.1%	85.9%
cities	26.5%	58.4%	79.5%	82.8%
OSs	38.4%	52.3%	78.8%	79.6%
cars	21.1%	32.2%	77.5%	78.7%

A comparative evaluation of a set of subsumption knowledge resources against the resulting integration matrix showed that the proposed techiniques enables better polarity detection. We plan to extend such evaluation to other kinds of knowledge (besides subsumption) and to more domains (besides opinion mining). Finally, further research studies are planned to investigate if a better trade-off between size and sparseness of the integration matrices can be achieved.

References

Bizer, C.; Jens, L.; Kobilarov, G.; Auer, S.; Becker, C.; Cyganiak, R.; and Hellmann, S. 2009. Dbpedia - a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web* 7(3):154–165.

Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; and Taylor, J. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *SIGMOD*, 1247–1250.

Cambria, E., and White, B. 2014. Jumping NLP curves: A review of natural language processing research. *IEEE Computational Intelligence Magazine* 9(2).

Cambria, E.; Hussain, A.; Havasi, C.; and Eckl, C. 2009. Common sense computing: From the society of mind to digital intuition and beyond. In Fierrez, J.; Ortega, J.; Esposito, A.; Drygajlo, A.; and Faundez-Zanuy, M., eds., *Biometric ID Management and Multimodal Communication*, volume 5707 of *Lecture Notes in Computer Science*. Berlin Heidelberg: Springer. 252–259.

Cambria, E.; Mazzocco, T.; Hussain, A.; and Eckl, C. 2011. Sentic medoids: Organizing affective common sense knowledge in a multi-dimensional vector space. In Liu, D.; Zhang, H.; Polycarpou, M.; Alippi, C.; and He, H., eds., *Advances in Neural Networks*, volume 6677 of *Lecture Notes in Computer Science*, 601–610. Berlin: Springer-Verlag.

Cambria, E.; Poria, S.; Gelbukh, A.; and Kwok, K. 2014. Sentic API: A common-sense based API for concept-level sentiment analysis. In *WWW*.

Carlson, A.; Betteridge, J.; Kisiel, B.; Settles, B.; Hruschka, E.; and Mitchell, T. 2010. Toward an architecture for never-ending language learning. In *AAAI*, 1306–1313.

Kuo, Y., and Hsu, J. 2012. Planning for reasoning with multiple common sense knowledge bases. *ACM Transactions on Interactive Intelligent Systems* 2(3):1–24.

Lenat, D., and Guha, R. 1989. *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Boston: Addison-Wesley.

Shank, R., and Tesler, L. 1969. A conceptual dependency parser for natural language. In *COLING*.

Speer, R., and Havasi, C. 2012. ConceptNet 5: A large semantic network for relational knowledge. In Hovy, E.; Johnson, M.; and Hirst, G., eds., *Theory and Applications of Natural Language Processing*. Springer. chapter 6.

Suchanek, F.; Kasneci, G.; and Weikum, G. 2007. Yago: a core of semantic knowledge. In *WWW*, 697–706.

Wu, W.; Li, H.; Wang, H.; and Zhu, K. 2012. Probase: A probabilistic taxonomy for text understanding. In *SIGMOD*, 481–492.