

## Chapter 2

# Background

*The good opinion of mankind,  
like the lever of Archimedes,  
with the given fulcrum, moves the world.*  
Thomas Jefferson

The World Wide Web represents one of the most revolutionary applications in the history of computing and human communication, which is keeping on changing how information is disseminated and retrieved, how business is conducted and how people communicate with each other. As the dimension of the Web increases, the technologies used in its development and the services provided to its users are developing constantly. Even if just few years have passed, in fact, Web 1.0's static and read-only HTML pages seem now just an old memory. Today the Web has become a dynamic and interactive reality in which more and more people actively participate by creating, sharing, and consuming contents. In this way, the World Wide Web configures itself not only as a 'Web of data', but also as a 'Web of people' where data and users are interconnected in an unbreakable bond.

This chapter shows how and why online opinions are important in the Web 2.0 era (Sect. 2.1) and illustrates existing approaches and depths of analysis in mining and characterising such opinions (Sect. 2.2). Eventually, the chapter comprises a background section on common sense knowledge representation and reasoning, which later will be further developed and applied to go beyond merely syntactical approaches to sentiment analysis (Sect. 2.3), and some concluding remarks (Sect. 2.4).

### 2.1 Opinion Mining and Sentiment Analysis

The passage from a read-only to a read-write Web made users more enthusiastic about interacting, sharing, and collaborating through social networks, online communities, blogs, wikis, and other online collaborative media. In the last years, this collective intelligence has spread to many different areas of the Web, with particular focus on fields related to our everyday life such as commerce, tourism, education, and health. The online review of commercial services and products, in particular, is an action that users usually perform with pleasure, to share their opinions about services they

have received or products they have just bought, and it constitutes immeasurable value for other potential buyers.

This trend opened new doors to enterprises that want to reinforce their brand and product presence in the market by investing in online advertising and positioning, that is, in social media marketing. The reasons why opinion mining is attracting so much attention from both the academic and the business world, in particular, can be found in the dynamics behind the buzz mechanism (Sect. 2.1.1), in the motivating factors that gave birth to the field (Sect. 2.1.2), and in the sub-tasks that make it different from standard information retrieval (Sect. 2.1.3).

### ***2.1.1 The Buzz Mechanism***

What mainly makes social media marketing work is the buzz mechanism [1]. A buzz replicates a message through user-to-user contact, rather than purchasing some advertising or promoting a press release. The message does not have to necessarily deal with the product. Many successful viral campaigns, in fact, have spread thanks to a compelling message, with the company logo included incidentally. At the heart of buzz is an understanding that the natural, spontaneous networks that comprise the social universe are the most effective means of reaching people in a meaningful way. The power of marketing lies, therefore, not in pushing information to the masses, but in effectively tapping those individuals who wield influence over others.

The marketers who are winning are the ones using consumers and culture to their advantage, crafting messages with consumers rather than throwing messages at them. In confirmation of the growing interest in this novel approach to marketing, several academic and commercial tools, e.g., OASYS<sup>1</sup> [2], ESSE [3], Luminoso<sup>2</sup> [4], Factiva,<sup>3</sup> NM Incite,<sup>4</sup> Attensity,<sup>5</sup> and Converseon,<sup>6</sup> have been developed to provide companies (and users) with a way to analyse the blogosphere on a large scale, in order to extract information about the trend of the opinions relative to their products. Nevertheless most of the existing tools and the research efforts are limited to a polarity evaluation or a mood classification according to a very limited set of emotions.

In addition, such methods mainly rely on parts of text in which emotional states are explicitly expressed and, hence, are unable to capture opinions and sentiments that are expressed implicitly.

---

<sup>1</sup> <http://oasys.umiacs.umd.edu/oasys>

<sup>2</sup> <http://lumino.so>

<sup>3</sup> <http://dowjones.com/factiva>

<sup>4</sup> <http://nmincite.com>

<sup>5</sup> <http://attensity.com>

<sup>6</sup> <http://converseon.com>

### 2.1.2 *Origins and Peculiarities*

The term ‘opinion mining’ first appears in a paper by Dave et al. [5] dated 2003, which envisioned the ideal opinion mining tool as capable to “process a set of search results for a given item, generating a list of product attributes (quality, features, etc.) and aggregating opinion about each of them (poor, mixed, good)”. From this early definition, the term opinion mining has been later extended to refer more generally to the computational techniques for extracting, classifying, understanding, and assessing the opinions expressed in various online news sources, social media comments, and other user-generated contents (UGCs). The introduction of the term ‘sentiment’ to the automatic analysis of evaluative text and tracking of the predictive judgements was first introduced in 2001 by Das and Chen [6] and Tong [7], in the context of market sentiment analysis.

In the context of NLP, the term sentiment can be intended either as the emotions or the polarity conveyed by text. Strictly speaking, sentiment analysis consists in inferring affective information from text, while opinion mining mainly concerns polarity detection. However, since the identification of sentiment, affect, subjectivity, and other emotional states is often propaedeutic to polarity detection [8], opinion mining and sentiment analysis are strictly connected and, therefore, commonly used interchangeably to denote the same field of study.

The manifesto of opinion mining and sentiment analysis as a unified field can be seen in the extensive review paper published by Pang and Lee [9] in 2008. This survey covers techniques and approaches that promise to directly enable opinion-oriented information-seeking systems. The authors’ focus is on methods that seek to address the new challenges raised by sentiment-aware applications, as compared to those that are already present in more traditional fact-based analysis. They include material on summarisation of evaluative text and on broader issues regarding privacy, manipulation, and economic impact that the development of opinion-oriented information-access services gives rise to.

To the inexperienced eye, opinion mining and sentiment analysis might look like the same as fields such as traditional text mining or fact-based analysis. Moreover, since sentiment classification deals with a relatively small number of classes, it might look like an easy task compared to text auto-categorisation. Opinion mining, however, is a very complex task even at its more basic level of sentiment polarity classification, which is a case of binary classification. The extraction of opinion polarity from text can be performed by comparing words extracted from text with a set of keywords with positive valence (e.g., love, wonderful, best, great, superb, still, beautiful) and negative valence (e.g., bad, worst, stupid, waste, boring), as in the case of topic-based binary classification. The identification of a right set of keywords for mining opinions, however, is not a trivial task. Even when machine learning techniques are employed to select keywords from training corpora, the level of accuracy is still very low if compared to the performance of typical topic-based binary classification [10]. The main reason is that, differently from topics, sentiments can often be expressed

in a more subtle manner, making it difficult to be identified by any of a sentence or document's terms when considered in isolation.

In addition, sentiment and subjectivity are quite context and domain dependent. This is true not only for changes in vocabulary, but also because even the exact same expression can indicate different sentiment in different domains. The concept 'go read the book', for example, most likely indicates positive sentiment for book reviews, but negative sentiment for movie reviews; as well as the adjective unpredictable may have a negative orientation in a car review (e.g., 'unpredictable steering'), but it could have a positive orientation in a movie review (e.g., 'unpredictable plot').

### 2.1.3 Sub-Tasks

One of the most common sub-tasks of opinion mining is polarity classification and the assignment of degrees of positivity, that is, given an opinionated piece of text wherein it is assumed that the overall opinion is about one single issue or item, classify the opinion as falling under one of two opposing sentiment polarities, or locate its position on the continuum between these two polarities. Much work on sentiment polarity classification has been conducted in the context of reviews of evaluative opinions (e.g., 'thumbs up' versus 'thumbs down', or 'like' versus 'dislike').

In addition, polarity classification can be also applied to identifying 'pro and con' expressions that can be used in individual reviews to evaluate the pros and cons that have influenced the judgements of a product and that make such judgements more trustworthy. Another instance of binary sentiment classification is agreement detection, that is, given a pair of text documents, deciding whether they should receive the same or differing sentiment-related labels. The more general problem of rating inference, where one must determine the author's evaluation with respect to a multi-point scale (e.g., one to five stars for a review) can be viewed as a multi-class text categorisation problem.

Other common sub-tasks of opinion mining and sentiment analysis are subjectivity detection and opinion identification. The capability of distinguishing if a text, or parts of it, are subjective or objective can be particularly beneficial for a more effective sentiment classification. Mihalcea et al. showed evidence that the complexity of this task is superior than subsequent polarity classification [11]. Wilson et al. remarked how classifying a piece of text as expressing a neutral opinion for rating inference does not equal classifying that piece of text as objective [12].

A piece of text can also have a polarity without necessarily containing an opinion, for example a news article can be classified into good or bad news without being subjective. The classification of a piece of text as subjective or objective can be useful in several situations. For example, being able to distinguish in opinionated texts where the authors do explicitly express their sentiment through statements (e.g., "this laptop is great") and where they provide objective information (e.g., "the laptop has long battery life") is used to help determine the overall sentiment. Hatzivassiloglou and Wiebe examined the effects of adjective orientation and gradability on sentence

subjectivity to detect if a sentence is subjective [13] while other projects address subjectivity detection at sub-sentence level. Wiebe et al. presented a comprehensive survey of subjectivity recognition using different clues and features [14].

Typically, sentiment analysis is performed over an on-topic document, e.g., on the result of a topic-based search engine. However, several studies suggested that managing these two task jointly can be beneficial for the overall performance. According to Riloff et al., topic-based text filtering and subjectivity filtering are complementary, in the context of experiments in information extraction [15]. For example, off-topic passages of a document could contain irrelevant affective information and result misleading for the global sentiment polarity about the main topic. Also, a document can contain material on multiple topics that may be of interest to the user. In this case, it is therefore necessary to identify the topics and separate the opinions associated with each of them. Several other researches in sentiment analysis focus on non-topic based categorisation, for example to classify documents according to their genre [16] and their style [17]. Also authorship and publisher identification are other relevant examples [18, 19]. Another problem that has been considered in intelligence and security settings is the detection of deceptive language. Affect detection, eventually, is also a task that is gaining a growing attention from different perspectives and for different applications.

Sentiment analysis has been traditionally more focused on the extraction of the valence of textual sample (i.e., positive/negative or bad/good) rather than assigning a particular emotion category to text. However, the classification of multimedia resources according to their mood and emotional content is also quite common. The advent of Web 2.0 has pushed the users at the centre of the Web universe, providing them revolutionary tools that have changed the way people communicate and express themselves, their ideas, and emotions. People spend more and more time using the Web not only for work, but also for expressing their opinions on blogs and forums, chatting, and organising events through social networks, and even for living a Second Life.<sup>7</sup> Therefore, the Web contains more and more affective content. The awareness that the capability to manage such affective content can be exploited for the development of next-generation web applications is dragging a growing attention also in sentiment analysis for affect extraction from textual Web content.

## 2.2 Main Approaches to Opinion Mining

Several approaches have been developed for the general task of mapping a given piece of text to a label belonging to a predefined set of categories, or to a real number representative of a polarity degree. Such approaches and their performance, however, are strictly bound to the considered domain of application and to the related topics. Moreover, most of the literature on sentiment analysis has focused on text written in English and consequently most of the resources developed, such as lexicons with sentiment labels, are in English. Adapting such resources to other languages

---

<sup>7</sup> <http://secondlife.com>

can be considered as a domain adaptation problem [20]. This section discusses the evolution of different approaches from heuristics to discourse structure (Sect. 2.2.1), from coarse to fine grained analysis (Sect. 2.2.2), from keyword to concept level opinion mining (Sect. 2.2.3).

### 2.2.1 From Heuristics to Discourse Structure

Several unsupervised learning approaches rely on the creation of a sentiment lexicon in an unsupervised manner that is later used to determine the degree of positivity (or subjectivity) of a text unit. The crucial component is, therefore, the creation of the lexicon via the unsupervised labelling of words or phrases with their sentiment polarity or subjectivity [9]. This lexicon can be used to identify the *prior polarity* or the *prior subjectivity* of terms or phrases, to use towards further identifying contextual polarity or subjectivity. Early works were mainly based on linguistic heuristics. For example, Hatzivassiloglou and McKeown’s technique [21] was built on the fact that, in the case of polarity classification, the two classes of interest represent opposites, and ‘opposition constraints’ can be used to help labelling decisions.

Other works propagated the valence of seed words, for which the polarity is known, to terms that co-occur with them in general text or in dictionary glosses, or to synonyms and words that co-occur with them in other WordNet-defined relations. A collective labelling approach can also be applied to opinion about product features. Popescu and Etzioni [22] proposed an iterative algorithm that, starting from a global word label computed over a large collection of generic topic text, gradually tried to re-define such label first to one that is specific to a review corpus then to one that is specific to a given product feature, and finally to one that is specific to the particular context in which the word occurs.

Also Snyder and Barzilay [23] exploited the idea of utilising discourse information to aid the inference of relationships between product attributes. They designed a linear classifier for predicting whether all aspects of a product are given the same rating, and combined such prediction with that of individual-aspect classifiers, in order to minimise a certain loss function. Regression techniques are often employed for the prediction of the degree of positivity in opinionated documents such as product reviews. Regression, in fact, allows to implicitly model similarity relationships between classes that correspond to points on a scale, such as the number of ‘stars’ given by a reviewer [9]. Modelling discourse structure, such as twists and turns in documents, contributes to a more effective overall sentiment labelling. Early works attempted to partially address this problem via incorporating location information in the feature set [24]. More recent studies have underlined that position is particularly relevant in the context of sentiment summarisation. In particular, in contrast to topic-based text summarisation, where the incipits of articles usually serve as a strong baseline, the last  $n$  sentences of a review have been shown to serve as a much better summary of the overall sentiment of the document, and to be almost as good as the  $n$  (automatically-computed) most subjective sentences [24]. Joshi and Rose

[25], for example, explored how features based on syntactic dependency relations can be utilised to improve performance on opinion mining. Using a transformation of dependency relation triples, they convert them into ‘composite back-off features’ that generalise better than the regular lexicalised dependency relation features.

### ***2.2.2 From Coarse to Fine Grained***

The evolution of research works in the field of opinion mining and sentiment analysis can be seen not only in the use of more and more sophisticated techniques, but also in the different depths of analysis adopted. Early works, in fact, aimed to classify entire documents as containing overall positive or negative polarity [10] or rating scores (e.g., 1–5 stars) of reviews [26]. These were mainly supervised approaches relying on manually labelled samples, such as movie or product reviews where the opinionist’s overall positive or negative attitude was explicitly indicated. However, opinions and sentiments do not occur only at document level, nor are they limited to a single valence or target. Contrary or complementary attitudes toward the same topic or multiple topics can be present across the span of a document. Later works adopted a segment level opinion analysis aiming to distinguish sentimental from non-sentimental sections, e.g., by using graph-based techniques for segmenting sections of a document on the basis of their subjectivity [24], or by performing a classification based on some fixed syntactic phrases that are likely to be used to express opinions [27], or by bootstrapping using a small set of seed opinion words and a knowledge base such as WordNet [28].

In recent works, text analysis granularity has been taken down to sentence level, e.g., by using presence of opinion-bearing lexical items (single words or n-grams) to detect subjective sentences [29, 30], or by using semantic frames defined in FrameNet [31] for identifying the topics (or targets) of sentiment [32], or by exploiting association rule mining [33] for a feature-based analysis of product reviews [34]. Commonly, a certain degree of continuity exists in subjectivity labels of adjacent sentences, as an author usually does not switch too frequently between being subjective and being objective. Hence, some works also propose a collective classification of the document based on assigning preferences for pairs of nearby sentences [26, 35].

All such approaches, however, are still far from being able to infer the cognitive and affective information associated with natural language as they mainly rely on semantic knowledge bases which are still too limited to efficiently process text at sentence level. Moreover, such a text analysis granularity might still not be enough as a single sentence may express more than one opinion [12].

### 2.2.3 From Keywords to Concepts

Existing approaches can be grouped into three main categories, with few exceptions: keyword spotting, lexical affinity, and statistical methods. Keyword spotting is the most naïve approach and probably also the most popular because of its accessibility and economy. Text is classified into affect categories based on the presence of fairly unambiguous affect words like ‘happy’, ‘sad’, ‘afraid’, and ‘bored’. Elliott’s Affective Reasoner [36], for example, watches for 198 affect keywords, e.g., ‘distressed’ and ‘enraged’, plus affect intensity modifiers, e.g., ‘extremely’, ‘somewhat’, and ‘mildly’, plus a handful of cue phrases, e.g., ‘did that’ and ‘wanted to’. Other popular sources of affect words are Ortony’s Affective Lexicon [37], which groups terms into affective categories, and Wiebe’s linguistic annotation scheme [38]. The weaknesses of this approach lie in two areas: poor recognition of affect when negation is involved and reliance on surface features. About its first weakness, while the approach can correctly classify the sentence “today was a happy day” as being happy, it is likely to fail on a sentence like “today wasn’t a happy day at all”. About its second weakness, the approach relies on the presence of obvious affect words which are only surface features of the prose.

In practice, a lot of sentences convey affect through underlying meaning rather than affect adjectives. For example, the text “My husband just filed for divorce and he wants to take custody of my children away from me” certainly evokes strong emotions, but uses no affect keywords, and therefore, cannot be classified using a keyword spotting approach. Lexical affinity is slightly more sophisticated than keyword spotting as, rather than simply detecting obvious affect words; it assigns arbitrary words a probabilistic ‘affinity’ for a particular emotion. For example, ‘accident’ might be assigned a 75 % probability of being indicating a negative affect, as in ‘car accident’ or ‘hurt by accident’. These probabilities are usually trained from linguistic corpora [39–42].

Though often outperforming pure keyword spotting, there are two main problems with the approach. First, lexical affinity, operating solely on the word-level, can easily be tricked by sentences like “I avoided an accident” (negation) and “I met my girlfriend by accident” (other word senses). Second, lexical affinity probabilities are often biased toward text of a particular genre, dictated by the source of the linguistic corpora. This makes it difficult to develop a reusable, domain-independent model. Statistical methods, such as latent semantic analysis (LSA) and support vector machine (SVM), have been popular for affect classification of texts and have been used by researchers on projects such as Goertzel’s Webmind [43], Pang’s movie review classifier [10], and many others [26, 34, 44–48]. By feeding a machine learning algorithm a large training corpus of affectively annotated texts, it is possible for the systems to not only learn the affective valence of affect keywords as in the keyword spotting approach, but such a system can also take into account the valence of other arbitrary keywords (like lexical affinity), punctuation, and word co-occurrence frequencies.



However, statistical methods are generally semantically weak, meaning that, with the exception of obvious affect keywords, other lexical or co-occurrence elements in a statistical model have little predictive value individually. As a result, statistical text classifiers only work with acceptable accuracy when given a sufficiently large text input. So, while these methods may be able to affectively classify user's text on the page or paragraph level, they do not work well on smaller text units such as sentences.

## 2.3 Towards Machines with Common Sense

Communication is one of the most important aspects of human life. Communicating has always a cost in terms of energy and time, since information needs to be encoded, transmitted, and decoded, and sometimes such factors can even make the difference between life and death. This is why people, when communicating with each other, provide just the useful information and take the rest for granted. This 'taken for granted' information is what we call common sense—obvious things people normally know and usually leave unstated. Common sense is not the kind of knowledge that we can find in Wikipedia,<sup>8</sup> but it consists in all the basic relationships among words, concepts, phrases, and thoughts that allow people to communicate with each other and face everyday life problems. It is a kind of knowledge that sounds obvious and natural to us, but it is actually daedal and multi-faceted.

The illusion of simplicity comes from the fact that, as each new group of skills matures, we build more layers on top of them and tend to forget about the previous layers. Common sense, in fact, is not a simple thing. Instead, it is an immense society of hard-earned practical ideas, of multitudes of life-learned rules and exceptions, dispositions and tendencies, balances and checks [49]. This section discusses the importance of common sense for the development of intelligent systems (Sect. 2.3.1) and illustrates different knowledge representation strategies (Sect. 2.3.2). The section also refers to a recently proposed survey on common sense computing [50] to present the evolution of such research field, from logic-based approaches to more recent methods based on natural language techniques (Sect. 2.3.3).

### 2.3.1 The Importance of Common Sense

Concepts are the glue that holds our mental world together [51]. Without concepts, there would be no mental world in the first place [52]. Doubtless to say, the ability to organise knowledge into concepts is one of the defining characteristics of human mind. Of the different sorts of semantic knowledge that are researched, arguably the most general and widely applicable kind is knowledge about the everyday world

---

<sup>8</sup> <http://wikipedia.org>

that is possessed by all people, i.e., common sense knowledge. While to the average person the term common sense is regarded as synonymous with good judgement, to the AI community it is used in a technical sense to refer to the millions of basic facts and understandings possessed by most people, e.g., “a lemon is sour”, “to open a door, you must usually first turn the doorknob”, “if you forget someone’s birthday, they may be unhappy with you”.

Common sense knowledge, thus defined, spans a huge portion of human experience, encompassing knowledge about the spatial, physical, social, temporal, and psychological aspects of typical everyday life. Because it is assumed that every person possesses common sense, such knowledge is typically omitted from social communications, such as text. A full understanding of any text then, requires a surprising amount of common sense, which currently only people possess. Common sense knowledge is what we learn and what we are taught about the world we live in during our formative years, in order to better understand and interact with the people and the things around us. Common sense is not universal, but cultural and context dependent. The importance of common sense can be particularly appreciated when travelling to far away places, where sometimes it is necessary to almost entirely reset oneself’s common sense knowledge in order to get integrated.

Despite the language barrier, in fact, moving to a new place involves facing habits and situations that might go against what we consider basic rules of social interaction or things we were taught by our parents, such as eating with hands, eating from someone else’s plate, slurping on noodle-like food or while drinking tea, eating on the street, crossing the road despite the heavy traffic, squatting when tired, removing shoes at home, growing long nails on your last fingers, or bargaining on anything you need to buy. This can happen also the other way around, that is, when you do something perfectly in line with your common sense that violates the local norms, e.g., cheek kissing as a form of greeting.

Common sense is the knowledge (usually acquired in early stages of our lives) concerning all the social, political, economic, and environmental aspects of the society we live in. Machines, as they never got the chance to live a life, have no common sense at all and, hence, they know nothing about us. To help us work, computers must get to know what our jobs are. To entertain us, they need to know what we like. To take care of us, they have to know how we feel. To understand us, they must think as we think. Today, in fact, computers do only what they are programmed to do. They only have one way to deal with a problem and, if something goes wrong, they get stuck. Nowadays we have programs that exceed the capabilities of world experts, but are not one able to do what a three years old child can do. It is because machines have no goals, no hopes, no fears; they do not know the meaning of things.

Computers can only do logical things, but meaning is an intuitive process—it cannot be reduced to zeros and ones. We need to transmit to computers our common sense knowledge of the world because soon there will not be enough human workers to perform the necessary tasks for our rapidly ageing population. To face this AI emergency,<sup>9</sup> we will have to give them physical knowledge of how objects behave,

---

<sup>9</sup> <http://mitworld.mit.edu/video/484>

social knowledge of how people interact, sensory knowledge of how things look and taste, psychological knowledge about the way people think, and so on. But having a database of millions of common sense facts will not be enough: we will also have to teach computers how to handle this knowledge, retrieve it when necessary, learn from experience—in a word, we will have to give them the capacity for common sense reasoning.

### ***2.3.2 Knowledge Representation***

From its very beginning, AI has rested on a foundation of formal representation of knowledge. Knowledge representation (KR) is a research area that directly addresses languages for representation and the inferences that go along with them. One of the central questions of KR research is in what form knowledge is to be expressed. One of the most popular representation strategies is first order logic (FOL), a deductive system that consists of axioms and rules of inferences and can be used to formalise relationally rich predicates and quantification [53].

FOL supports syntax, semantics and, to a certain degree, pragmatics expressions. Syntax specifies the way groups of symbols are to be arranged, so that the group of symbols is considered properly formed. Semantics specify what well-formed expressions are supposed to mean. Pragmatics specifies how contextual information can be leveraged to provide better correlation between different semantics, for tasks such as word sense disambiguation. Logic, however, is known to have the problem of monotonicity. The set of entailed sentences can only increase as information is added to the knowledge base. This violates a common property of human reasoning, i.e., changing one's mind. Solutions such as default and linear logic serve to address parts of these issues. Default logic is proposed by Raymond Reiter to formalise default assumptions, e.g., “all birds fly” [54]. However, issues arise when default logic formalise facts that are true in the majority of cases, but not always, e.g., “penguins do not fly”.

Linear logic, or constructive logic, was developed by Arend Heyting [55]. It is a symbolic logical system that preserves justification, rather than truth, and supports rejecting the weakening and contraction rules. It excels in careful deductive reasoning and is suitable in situations that can be posed precisely. As long as a scenario is static and can be detailedly described, in fact, situation-specific rules can perfectly model it but, when it comes to capture a dynamic and uncertain real-world environment, logical representation usually fails for lack of generalisation capabilities. Accordingly, it is not natural for human to encode knowledge in logical formalisation. Another standard KR strategy, based on FOL, is the use of relational databases. The idea is to describe a database as a collection of predicates over a finite set of variables and describing constraints on the possible values. Structured query language (SQL) [56] is the database language designed for the retrieval and management of data in relational

database management systems (RDBMS) [57]. Commercial (e.g., Oracle,<sup>10</sup> Sybase,<sup>11</sup> Microsoft SQL Server<sup>12</sup>) and open-source (e.g., MySQL<sup>13</sup>) implementations of RDBMS are available and they are commonly used in the IT industry.

Relational database design requires a strict process called normalisation to ensure that the relational database is suitable for general purpose querying and the relational database is free of database operations anomalies. Third normal form (3NF) [58] is stricter than first and second normal forms and less strict as compared to Boyce-Codd normal form (BCNF) [59], fourth, and fifth normal forms. Stricter normal forms means that the database design is more structured and, hence, requires more database tables. The advantage is that the overall design looks more organised. The disadvantage is the performance trade-off when joint table SQL queries are invoked. Relational database design, moreover, does not directly address representation of parent-child relationship in the object-oriented paradigm, subjective degrees of confidence, and temporal dependent knowledge.

A popular KR strategy, especially among Semantic Web researchers, is production rule [60]. A production rule system keeps a working memory of on-going memory assertions. This working memory is volatile and keeps a set of production rules. A production rule comprises an antecedent set of conditions and a consequent set of actions (i.e., IF {conditions} THEN {actions}). The basic operation for a production rule system involves a cycle of three steps ('recognise', 'resolve conflict', and 'act') that repeats until no more rules are applicable to working memory. The step 'recognise' identifies the rules whose antecedent conditions are satisfied by the current working memory. The set of rules identified is also called the conflict set. The step 'resolve conflict' looks into the conflict set and selects a set of suitable rules to execute. The step 'act' simply executes the actions and updates the working memory. Production rules are modular. Each rule is independent from others, allowing rules to be added and deleted easily.

Production rule systems have simple control structure and the rules are easy for human to understand. This is because rules are usually derived from observation of expert behaviour or expert knowledge, thus the terminology used in encoding the rules tend to resonate with human understanding. However, there are issues with scalability when production rule systems get larger. Significant maintenance overhead is required to maintain systems with thousands of rules.

Another prominent KR strategy among Semantic Web researchers is the ontology web language (OWL),<sup>14</sup> an XML-based vocabulary that extends resource description framework (RDF)<sup>15</sup> and resource description framework schema (RDFS)<sup>16</sup> to provide a more comprehensive ontology representation, such as the definition of classes,

---

<sup>10</sup> <http://oracle.com>

<sup>11</sup> <http://sybase.com>

<sup>12</sup> <http://microsoft.com/sqlserver>

<sup>13</sup> <http://mysql.com>

<sup>14</sup> <http://w3.org/TR/owl-overview>

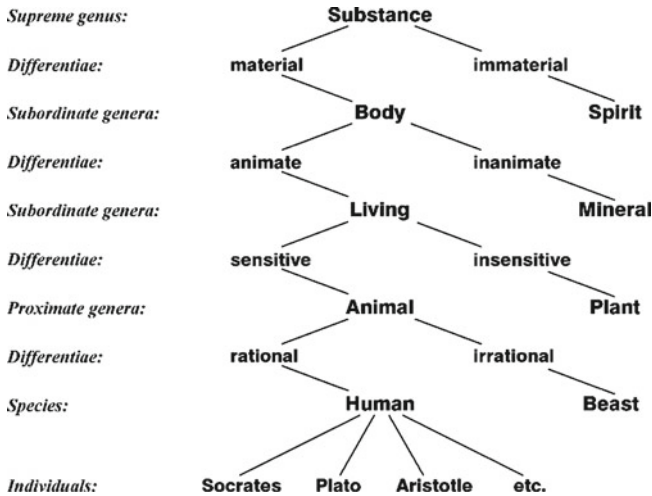
<sup>15</sup> <http://w3.org/TR/PR-rdf-syntax>

<sup>16</sup> <http://w3.org/2001/sw/wiki/RDFS>

relationships between classes, properties of classes, and constraints on relationships between classes and properties of classes. RDF supports subject-predicate-object model that makes assertion about a resource. Reasoning engines have been developed to check for semantic consistency and help to improve ontology classification. OWL is a W3C recommended specification and comprises three dialects: OWL-Lite, OWL-DL, and OWL-Full. Each dialect has a different level of expressiveness and reasoning capabilities. OWL-Lite is the least expressive compared to OWL-Full and OWL-DL. It is suitable for building ontologies that only require classification hierarchy and simple constraints and, for this reason, it provides the most computationally efficient reasoning. OWL-DL is more expressive than OWL-Full, but more expressive than OWL-Lite. It has restrictions on the use of some of the description tags, hence, computation formed by a reasoning engine on OWL-DL ontologies can be completed in a finite amount of time [61]. OWL-DL is so named due to its correspondence with description logic. It is also the most commonly used dialect for representing domain ontology for Semantic Web applications. OWL-Full is the complete language and is useful for modelling a full representation of a domain. However, the trade-off for OWL-Full is the high complexity of the model that can result in sophisticated computation that may not complete in finite time. In general, OWL requires strict definition of static structures, hence, it is not suitable for representing knowledge that requires subjective degrees of confidence, but rather for representing declarative knowledge. OWL, moreover, does not allow to easily represent temporal dependent knowledge.

Another well-known way to represent knowledge is to use networks. Bayesian networks [62], for example, provide a means of expressing joint probability distributions over many interrelated hypotheses. Bayesian network is also called belief network. All variables are represented using directed acyclic graph (DAG). The nodes of a DAG represent variables. Arcs are causal connections between two variables where the truth of the former directly affects the truth of the latter. A Bayesian network is able to represent subjective degrees of confidence. The representation explicitly explores the role of prior knowledge and combines evidence of the likelihood of events. In order to compute the joint distribution of the belief network, there is a need to know  $\Pr(P|\text{parents}(P))$  for each variable  $P$ . It is difficult to determine the probability of each variable  $P$  in the belief network. Hence, it is also difficult to scale and maintain the statistical table for large scale information processing problem. Bayesian networks also have limited expressiveness, which is only equivalent to the expressiveness of proposition logic. For this reason, semantic networks are more often used for KR (Fig. 2.1).

A semantic network [63] is a graphical notation for representing knowledge in patterns of interconnected nodes and arcs. There are six types of networks, namely definitional networks, assertional networks, implicational networks, executable networks, learning networks, and hybrid networks. A definitional network focuses on *IsA* relationships between a concept and a newly defined sub-type. The resulting network is called a generalisation, which supports the rule of inheritance for copying properties defined for a super-type to all of its sub-types. Definitions are true by definition and, hence, the information in definitional networks is often assumed to



**Fig. 2.1** Tree of Porphyry. Porphyry presented the basis of Aristotle’s thought as a tree-like scheme of dichotomous divisions, in which the process continues until the lowest species is reached

be true. Assertional networks are meant to assert propositions and the information is assumed to be contingently true. Contingent truth means that the proposition is true in some but not in all the worlds. The proposition also has sufficient reason in which the reason entails the proposition, e.g., “the stone is warm” with the sufficient reasons being “the sun is shining on the stone” and “whatever the sun shines on is warm”. Contingent truth is not the same as the truth that is assumed in default logic. Contingent truth is closer to the truth assumed in model logic.

Implicational networks use implication as the primary relation for connecting nodes. They are used to represent patterns of beliefs, causality, or inferences. Methods for realising implicational networks include Bayesian networks and logic inferences used in a truth maintenance system (TMS). By combinations of forward and backward reasoning, a TMS propagates truth-values to nodes whose truth-value is unknown.

Executable networks contains mechanisms implemented in run-time environment such as message passing, attached procedure (e.g., data-flow graph), and graph transformation that can cause change to the network. Learning networks acquire knowledge from examples by adding and deleting nodes and links, or by modifying weights associated with the links. Learning networks can be modified in three ways: rote memory, changing weights, and restructuring. As for the rote memory, the idea is to add information without making changes to the current network. Exemplar methods can be found in relational database. For example, Patrick Winston used a version of relational graphs to describe structures, such as arches and towers [64]. When his program was given positive and negative examples of each type of structure, it would generalise the graphs to derive a definitional network for classifying all the types of structures that were considered. The idea of changing weights, in turn, is to modify

the weights of links without changing the network structure for the nodes and links. Exemplar methods can be found in neural networks. As for restructuring, finally, the idea is to create fundamental changes to the network structure for creative learning. Methods include case-based reasoning. The learning system uses rote memory to store various cases and associated action such as course of action. When a new case is encountered, the system finds those cases that are most similar to the new one and retrieves the outcome. To organise the search and evaluate similarity, the learning system must use restructuring to find common patterns in the individual cases and use those patterns as the keys for indexing the database. Hybrid networks combine two or more of the previous techniques. Hybrid networks can be a single network. They can also be separate but closely interacting networks.

Sowa used unified modelling language (UML) as an example to illustrate a hybrid semantic network. Semantic networks are very expressive. The representation is flexible and can be used to express different paradigm such as relational model and hierarchical relationship. The challenge is at implementation level. For example, it is difficult to implement hybrid semantic network, which requires an integration of different methods.

### ***2.3.3 From Logical Inference to Digital Intuition***

What magical trick makes us intelligent?—Marvin Minsky was wondering more than two decades ago—The trick is that there is no trick. The power of intelligence stems from our vast diversity, not from any single, perfect principle [49]. Human brain, in fact, is a very complex system, maybe the most complex in nature. The functions it performs are the product of thousands and thousands of different subsystems working together at the same time. Common sense computing involves trying to emulate such mechanism and, in particular, exploiting common sense knowledge to improve computers' understanding of the world. Before Minsky, many AI researchers started to think about the implementation of a common sense reasoning machine.

The very first person who seriously started thinking about the creation of such a machine was perhaps Alan Turing when, in 1950, he first raised the question “can machines think?”. But he never managed to answer that question, he just provided a method to gauge artificial intelligence, the famous Turing test. The notion of common sense in AI is actually dated 1958, when John McCarthy, in his seminal paper ‘Programs with Common Sense’ [65], proposed a program, termed the ‘advice taker’, for solving problems by manipulating sentences in formal language. The main aim of such a program was to try to automatically deduce for itself a sufficiently wide class of immediate consequences of anything it was told and what it already knew. In this paper, McCarthy stressed the importance of finding a proper method of representing expressions in the computer since, in order for a program to be capable of learning something, it must first be capable of being told. He also developed the idea of creating a property list for each object, in which the specific things people usually know about that object are listed. It was the first attempt to build a common sense



knowledge base but, more important, it was the epiphany of the need of common sense to move forward in the technological evolution.

In 1959, McCarthy went to MIT and started, together with Minsky, the MIT Artificial Intelligence Project. They both were aware of the need for AI of a common sense reasoning approach, but while McCarthy was more concerned with establishing logical and mathematical foundations for it, Minsky was more involved with theories of how we actually reason using pattern recognition and analogy. These theories were organised some years later with the publication of the *Society of Mind* [49], a masterpiece of AI literature, which consists in an illuminating vision of how the human brain might work. Minsky sees the mind made of many little parts, termed ‘agents’, each mindless by itself but able to lead to true intelligence when working together. These groups of agents, called ‘agencies’, are responsible to perform some type of function, such as remembering, comparing, generalising, exemplifying, analogising, simplifying, predicting, and so on.

The most common agents are the so called ‘K-lines’, whose task is simply to activate other agents: this is a very important issue since agents are all highly interconnected and activating a K-line can cause a significant cascade of effects. To Minsky, in fact, mental activity ultimately consists in turning individual agents on and off: at any time only some agents are active and their combined activity constitutes the ‘total state’ of the mind. K-lines are a very simple but powerful mechanism since they allow entering a particular configuration of agents that formed a useful society in a past situation. This is how we build and retrieve our problem solving strategies in our mind; this is how we should develop our problem solving strategies in our programs.

In 1990, McCarthy put together seventeen papers to try to define common sense knowledge by using mathematical logic in such a way that common sense problems could be solved by logical reasoning. Deductive reasoning in mathematical logic has the so-called monotonicity property: if we add new assumptions to the set of initial assumptions, there may be some new conclusions, but every sentence that was a deductive consequence of the original hypotheses is still a consequence of the enlarged set.

Much human reasoning is monotonic as well, but some important human common sense reasoning is not. For example, if someone is asked to build a birdcage, this person concludes that it is appropriate to put a top on it, but when he/she learns the further fact that the bird is a penguin he/she no longer draws that conclusion. McCarthy formally described this assumption that things are as expected unless otherwise specified, with the ‘circumscription method’ of non-monotonic reasoning: a minimisation similar to the closed world assumption that what is not known to be true is false. In the same years, a similar attempt to give a shape to common sense knowledge was done by Ernest Davis [66]. He tried to develop an ad hoc language for expressing common sense knowledge and inference techniques for carrying out common sense reasoning in specific domains such as space, time, quantities, qualities, flows, goals, plans, needs, beliefs, intentions, actions, and interpersonal relations. Thanks to his and McCarthy’s knowledge formalisations, the first steps were set towards the expression of common sense facts in a way that would have been suitable



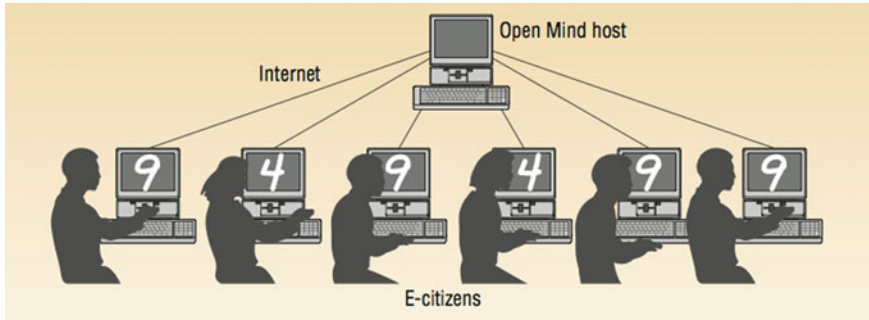
for inclusion in a general purpose database and, hence, towards the development of programs with common sense.

Minsky's theory of human cognition, in particular, was welcomed with great enthusiasm by the AI community and gave birth to many attempts to build common sense knowledge bases and develop systems capable of common sense reasoning. The most representative projects are Cyc [67], Doug Lenat's logic-based repository of common sense knowledge, WordNet [68], Christiane Fellbaum's universal database of word senses, and ThoughtTreasure [69], Erik Mueller's story understanding system. Cyc is one of the first attempts to assemble a massive knowledge base spanning human common sense knowledge.

Initially started by Doug Lenat in 1984, this project utilises knowledge engineers who hand-craft assertions and place them into a logical framework using CycL, Cyc's proprietary language. Cyc's knowledge is represented redundantly at two levels: a frame language distinction (epistemological level), adopted for its efficiency, and a predicate calculus representation (heuristic level), needed for its expressive power to represent constraints. While the first level keeps a copy of the facts in the uniform user language, the second level keeps its own copy in different languages and data structures suitable to be manipulated by specialised inference engines. Knowledge in Cyc is also organised into 'microtheories', resembling Minsky's agencies, each one with its own knowledge representation scheme and sets of assumptions. These microtheories are linked via 'lifting rules' that allow translation and communication of expressions between them.

Begun in 1985 at Princeton University, WordNet is a database of words (primarily nouns, verbs, and adjectives). It has been one of the most widely used resources in computational linguistics and text analysis for the ease in interfacing it with any kind of application and system. The smallest unit in WordNet is the word/sense pair, identified by a 'sense key'. Word/sense pairs are linked by a small set of semantic relations such as synonyms, antonyms, *IsA* superclasses, and words connected by other relations such as *PartOf*. Each synonym set, in particular, is called 'synset': it consists in the representation of a concept, often explained through a brief gloss, and represents the basic building block for hierarchies and other conceptual structures in WordNet. Erik Mueller's ThoughtTreasure is a story understanding system with a great variety of common sense knowledge about how to read and understand children's stories. It was inspired by Cyc and is similar to Cyc in that it has both natural language and common sense components. But whereas Cyc mostly uses logic, ThoughtTreasure uses multiple representations schemes: grids for stereotypical settings, finite automata for rules of device behaviour and mental processes, logical assertions for encyclopaedic facts and linguistic knowledge. ThoughtTreasure's lexicon is similar to WordNet but, while world knowledge is explicitly excluded from WordNet, ThoughtTreasure contains also concepts that are not lexicalised in English like 'going to the pub' or 'eating at the restaurant', which are very important for common sense reasoning.

Using logic-based reasoning, in fact, can solve some problems in computer programming, but most real-world problems need methods better at matching patterns and constructing analogies, or making decisions based on previous experience with



**Fig. 2.2** An Open Mind project on OCR: handwritten characters are presented to e-citizens whose judgements (here 4 versus 9) are returned to the Open Mind host and used to train the classifier

examples, or by generalising from types of explanations that have worked well on similar problems in the past [70]. In building intelligent systems we have to try to reproduce our way of thinking: we turn ideas around in our mind to examine them from different perspectives until we find one that works for us. From this arises the need of using several representations, each integrated with its set of related pieces of knowledge, to be able to switch from one to another when one of them fails. The key, in fact, is using different representations to describe the same situation.

Minsky blames our standard approach to writing a program for common sense computing failures. Since computers appeared, our approach to solve a problem has always consisted in first looking for the best way to represent the problem, and then looking for the best way to represent the knowledge needed to solve it and finally looking for the best procedure for solving it. This problem-solving approach is good when we have to deal with a specific problem, but there is something basically wrong with it: it leads us to write only specialised programs that cope with solving only that kind of problem. This is why, today, we have millions of expert programs but not even one that can be actually defined intelligent.

From here comes the idea of finding a heterogeneous ways to represent common sense knowledge and to link each unit of knowledge to the uses, goals, or functions that each knowledge-unit can serve. This non-monotonic approach reasserted by Minsky was adopted soon after by Push Singh within the Open Mind Common Sense (OMCS) project [71]. Initially born from an idea of David Stork [72], the project differs from previous attempts to build a common sense database for the innovative way to collect knowledge and represent it (Fig. 2.2). OMCS is a second-generation common sense database. Knowledge is represented in natural language, rather than using a formal logical structure, and information is not hand-crafted by expert engineers but spontaneously inserted by online volunteers. The reason why Lenat decided to develop an ad hoc language for Cyc is that vagueness and ambiguity pervade English and computer reasoning systems generally require knowledge to be expressed accurately and precisely. However, as expressed in the Society of Mind, ambiguity is unavoidable when trying to represent the common sense world.

No single argument, in fact, is always completely reliable but, if we combine multiple types of arguments, we can improve the robustness of reasoning as well as we can improve a table stability by providing it with many small legs in place of just one very big leg. This way information is not only more reliable, but also stronger. If a piece of information goes lost, we can still access the whole meaning, exactly as the table keeps on standing up if we cut out one of the small legs. Diversity is, in fact, the key of OMCS' success: the problem is not choosing a representation in spite of another, but it is finding a way for them to work together in one system. The main difference between acquiring knowledge from the general public and acquiring it from expert engineers is that the general public is likely to leave as soon as they encounter something boring or difficult. The key is letting people do what they prefer to do. Different people, in fact, like to do different things: some like to enter new items, some like to evaluate items, others like to refine items. For this reason, OMCS is based on a distributed workflow model where the different stages of knowledge acquisition could be performed separately by different participants. The system, in fact, was designed to allow users to insert new knowledge via both template-based input and free-form input, tag concepts, clarify properties, and validate assertions. But, since giving so much control to users can be dangerous, a fixed set of pre-validated sentences were meant to be presented to them from time to time, in order to assess their honesty, and the system was designed in a way that allowed users to reciprocally control each other by judging samples of each other's knowledge. OMCS exploits a method termed cumulative analogy [73], a class of analogy-based reasoning algorithms that leverage existing knowledge to pose knowledge acquisition questions to the volunteer contributors. When acquiring knowledge online, the stickiness of the website is of primary importance. The best way to involve users in this case is making them feel that they are contributing to the construction of a thinking machine and not just a static database. To do this, OMCS first determines what other topics are similar to the topic the user is currently inserting knowledge for, and then it uses cumulative analogy to generate and present new specific questions about this topic. Because each statement consists of an object and a property, the entire knowledge repository can be visualised as a large matrix, with every known object of some statement being a row and every known property being a column. Cumulative analogy is performed by first selecting a set of nearest neighbours, in terms of similarity, of the treated concept and then by projecting known properties of this set onto not known properties of the concept and presenting them as questions (Fig. 2.3). The replies to the knowledge acquisition questions formulated by analogy are immediately added to the knowledge repository, affecting the similarity calculations. This way users can see the system's behaviour improve or change as a result of the entered knowledge and be more tempted to participate.

A more generalised way to deal with the information contained in the Open Mind corpus is AnalogySpace [74], a process that applies singular value decomposition (SVD) on the matrix representation of the common sense knowledge base, in order to reduce its dimensionality and capture the most important correlations. The entries in the resulting matrix are positive or negative numbers, depending on the reliability of the assertions, and their magnitude increases logarithmically with the confidence

Objects	Properties (with simplified form)				
	...	contains knowledge <i>contain knowledge</i>	has pages <i>have page</i>	is cold <i>be cold</i>	is for reading <i>be read</i>
...	...	...	...	...	...
book	...	x	x		x
ice	...		-	x	
newspaper	...	x?	x		x
magazine	...	x	x		x
...	...	...	...	...	...

**Fig. 2.3** The cumulative analogy process allows to perform comparisons between concepts in a knowledge base (represented as a matrix) and, hence, to infer new information about similar concepts

score. Applying SVD on this matrix causes it to describe other features that could apply to known concepts by analogy: if a concept in the matrix has no value specified for a feature owned by many similar concepts, then by analogy the concept is likely to have that feature as well.

A way to visualise and understand AnalogySpace is provided by Luminoso [4], a tool that allows to interactively explore the dimensionality-reduced semantic space of common sense knowledge by ‘grabbing’ its data points and, hence, view their associated text and statistics. This operation also allows highlighting the point’s neighbourhood of semantically associated data points, providing clues for reasons as to why the points were classified along the dimensions they were.

The AnalogySpace process, eventually, is naturally extended by the ‘blending’ technique [75], a new method to perform inference over multiple sources of data simultaneously, taking advantage of the overlap between them. Blending consists in an alignment phase of two datasets and of a linear combination of them to be able to apply principal component analysis (PCA) on the resulting matrix. This enables common sense to be used as a basis for inference in a wide variety of systems and applications so that they can achieve digital intuition about their own data, making assumptions and conclusions based on the connections between that specific data and the general common sense that people have.

## 2.4 Conclusions

This chapter has shown how and why, today, online opinions are crucial both for companies to succeed in selling their products and services, and for individuals to properly choose among these. The dynamics behind the buzz mechanism were discussed, together with the motivating factors that gave birth to the field of opinion mining, and the tasks that make it different from standard information retrieval (Sect. 2.1). The chapter also illustrated the approaches and depths of analysis in mining and

characterising opinions, in order to map a given piece of text to a label belonging to a predefined set of categories, or to a real number representative of a polarity degree.

Specifically, the chapter discussed the evolution of different approaches from heuristics to discourse structure, from coarse to fine grained analysis, and from key-word to concept level opinion mining (Sect. 2.2). Eventually, the chapter explained the importance of common sense for the development of intelligent systems, illustrated different knowledge representation strategies, and presented the evolution of common sense computing from logic-based methods to more recent approaches based on natural language techniques (Sect. 2.3).

## References

1. Salzman, M., Matathia, I., O'Reilly, A.: *Buzz: Harness the Power of Influence and Create Demand*. Wiley, New York (2003)
2. Cesarano, C., Dorr, B., Picariello, A., Reforgiato, D., Sagoff, A., Subrahmanian, V.: OASYS: An opinion analysis system. AAAI CAAW. Stanford, In (2006)
3. Sood, S., Vasserman, L.: ESSE: exploring mood on the web. ICWSM. San Jose, In (2009)
4. Speer, R., Havasi, C., Treadway, N., Lieberman, H.: Finding your way in a multi-dimensional semantic space with Luminoso. IUI. Hong Kong, In (2010)
5. Dave, K., Lawrence, S., Pennock, D.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. WWW. Budapest, In (2003)
6. Das, S., Chen, M.: Yahoo! for Amazon: extracting market sentiment from stock message boards. APFA. Bangkok, In (2001)
7. Tong, R.: An operational system for detecting and tracking opinions in on-line discussion. SIGIR. New Orleans, In (2001)
8. Hsinchun, C., Zimbra, D.: AI and opinion mining. *IEEE Intell. Syst.* **25**(3), 74–80 (2010)
9. Pang, B., Lee, L.: Opinion mining and sentiment analysis. *Found. Trends Inf. Retrieval* **2**, 1–135 (2008)
10. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: *EMNLP*, pp. 79–86. Philadelphia (2002).
11. Mihalcea, R., Banea, C., Wiebe, J.: Learning multilingual subjective language via cross-lingual projections. *ACL*. Prague, In (2007)
12. Wilson, T., Wiebe, J., Hwa, R.: Just how mad are you? Finding strong and weak opinion clauses. In: *AAAI*, pp. 761–769. San Jose (2004).
13. Hatzivassiloglou, V., Wiebe, J.: Effects of adjective orientation and gradability on sentence subjectivity. *COLING*. Saarbrücken, In (2000)
14. Wiebe, J., Wilson, T., Bruce, R., Bell, M., Martin, M.: Learning subjective language. *Comput. Linguist.* **30**(3) (2004).
15. Riloff, E., Wiebe, J., Phillips, W.: Exploiting subjectivity classification to improve information extraction. *AAAI*. Pittsburgh, In (2005)
16. Finn, A., Kushmerick, N.: Learning to classify documents according to genre. *J. Am. Soc. Inf. Sci. Technol.* **7**(5) (2006).
17. Biber, D.: *Variation across Speech and Writing*. Cambridge University Press, Cambridge (1988)
18. Mosteller, F., Wallace, D.: *Applied Bayesian and Classical Inference: The Case of the Federalist Papers*. Springer, Berlin (1984)
19. Argamon, S., Koppel, M., Avneri, G.: Style-based text categorization: What newspaper am i reading?. *AAAI Workshop on Text Categorization*. Madison, In (1998)

20. Cambria, E., Hussain, A., Durrani, T., Wang, Q.: Towards a chinese common and common sense knowledge base for sentiment analysis. In: Jiang, H., Ding, W., Ali, M., Wu, X. (eds.) *Advanced Research in Applied Artificial Intelligence, Lecture Notes in Artificial Intelligence*, vol. 7345, pp. 437–446. Springer, Berlin (2012).
21. Hatzivassiloglou, V., McKeown, K.: Predicting the semantic orientation of adjectives. In: *ACL/EACL*. Madrid (1997).
22. Popescu, A., Etzioni, O.: Extracting product features and opinions from reviews. In: *HLT/EMNLP*. Vancouver (2005).
23. Snyder, B., Barzilay, R.: Multiple aspect ranking using the good grief algorithm. In: *HLT/NAACL*. Rochester (2007).
24. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: *ACL*, pp. 271–278. Barcelona (2004).
25. Joshi, M., Rose, C.: Generalizing dependency features for opinion mining. *ACL/IJCNLP*. Singapore, In (2009)
26. Pang, B., Lee, L.: Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In: *ACL*, pp. 115–124. Ann, Arbor (2005).
27. Turney, P.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: *ACL*, pp. 417–424. Philadelphia (2002).
28. Kamps, J., Marx, M., Mokken, R., de Rijke, M.: Using WordNet to measure semantic orientation of adjectives. In: *LREC*, pp. 1115–1118. Lisbon (2004).
29. Kim, S., Hovy, E.: Automatic detection of opinion bearing words and sentences. In: *IJCNLP*, pp. 61–66. Jeju Island (2005).
30. Riloff, E., Wiebe, J.: Learning extraction patterns for subjective expressions. In: *EMNLP*, pp. 105–112. Sapporo (2003).
31. Baker, C., Fillmore, C., Lowe, J.: The Berkeley FrameNet project. In: *COLING/ACL*, pp. 86–90. Montreal (1998).
32. Kim, S., Hovy, E.: Extracting opinions, opinion holders, and topics expressed in online news media text. *Workshop on Sentiment and Subjectivity in Text*. Sydney, In (2006)
33. Agrawal, R., Srikant, R.: Fast algorithm for mining association rules. *VLDB*. Santiago de Chile, In (1994)
34. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: *KDD*, Seattle (2004)
35. Zirn, C., Niepert, M., Stuckenschmidt, H., Strube, M.: Fine-grained sentiment analysis with structural features. *IJCNLP*. Chiang Mai, In (2011)
36. Elliott, C.D.: The affective reasoner: a process model of emotions in a multi-agent system. Ph.D. thesis, Northwestern University, Evanston (1992).
37. Ortony, A., Clore, G., Collins, A.: *The Cognitive Structure of Emotions*. Cambridge University Press, Cambridge (1988)
38. Wiebe, J., Wilson, T., Cardie, C.: Annotating expressions of opinions and emotions in language. *Lang. Resour. Eval.* **39**(2), 165–210 (2005)
39. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis. *HLT/EMNLP*. Vancouver, In (2005)
40. Stevenson, R., Mikels, J., James, T.: Characterization of the affective norms for english words by discrete emotional categories. *Behav. Res. Methods* **39**, 1020–1024 (2007)
41. Somasundaran, S., Wiebe, J., Ruppenhofer, J.: Discourse level opinion interpretation. In: *COLING*. Manchester (2008).
42. Rao, D., Ravichandran, D.: Semi-supervised polarity lexicon induction. In: *EACL*, pp. 675–682. Athens (2009).
43. Goertzel, B., Silverman, K., Hartley, C., Bugaj, S., Ross, M.: The Baby Webmind project. In: *AISB*. Birmingham (2000).
44. Turney, P., Littman, M.: Measuring praise and criticism: inference of semantic orientation from association. *ACM Trans. Inf. Syst.* **21**(4), 315–346 (2003)
45. Abbasi, A., Chen, H., Salem, A.: Sentiment analysis in multiple languages: feature selection for opinion classification in web forums. *ACM Trans. Inf. Syst.* **26**(3), 1–34 (2008)

46. Nguyen, L., Wu, P., Chan, W., Peng, W., Zhang, Y.: Predicting collective sentiment dynamics from time-series social media. In: ACM KDD WISDOM, Beijing (2012)
47. Di Fabbrizio, G., Aker, A., Gaizauskas, R.: Starlet: Multi-document summarization of service and product reviews with balanced rating distributions. In: IEEE ICDM SENTIRE, Vancouver (2011)
48. Velikovich, L., Goldensohn, S., Hannan, K., McDonald, R.: The viability of web-derived polarity lexicons. In: NAACL, pp. 777–785. Los Angeles (2010).
49. Minsky, M.: *The Society of Mind*. Simon and Schuster, New York (1986)
50. Cambria, E., Hussain, A., Havasi, C., Eckl, C.: Common sense computing: from the society of mind to digital intuition and beyond. In: Fierrez, J., Ortega, J., Esposito, A., Drygajlo, A., Faundez-Zanuy, M. (eds.) *Biometric ID Management and Multimodal Communication*, Lecture Notes in Computer Science, p. 259. Springer, Berlin (2009).
51. Murphy, G.: *The Big Book of Concepts*. The MIT Press, Cambridge (2004)
52. Bloom, P.: Glue for the mental world. *Nature* **421**, 212–213 (2003)
53. Barwise, J.: An introduction to first-order logic. *Handbook of Mathematical Logic*. In: *Studies in Logic and the Foundations of Mathematics*. North-Holland (1977).
54. Reiter, R.: A logic for default reasoning. *Artif. Intell.* **13**, 81–132 (1980)
55. Heyting, A.: *Intuitionism. An introduction* North-Holland (1956).
56. Date, C., Darwen, H.: *A Guide to the SQL Standard*. Addison-Wesley, Reading (1993).
57. Codd, E.: A relational model of data for large shared data banks. *Commun. ACM* **13**(6), 377–387 (1970)
58. Codd, E.: Further normalization of the data base relational model. Technical Report, IBM Research Report, New York (1971)
59. Codd, E.: Recent investigations into relational data base systems. Technical Report RJ1385, IBM Research Report, New York (1974).
60. Chomsky, N.: Three models for the description of language. *IRE Trans. Inf. Theory* **2**(3), 113–124 (1956)
61. Lacy, L.: *OWL: Representing Information Using the Web Ontology Language*. Trafford Publishing, Victoria (2005).
62. Pearl, J.: Bayesian networks: a model of self-activated memory for evidential reasoning. Technical Report CSD-850017, UCLA Technical Report, Irvine (1985).
63. Sowa, J.: Semantic networks. *Encyclopedia of Artificial Intelligence*. Stuart Shapiro, In (1987)
64. Winston, P.: Learning structural descriptions from examples. *The Psychology of Computer Vision*. pp. 157–209. McGraw-Hill, New York (1975).
65. McCarthy, J.: Programs with common sense. *Teddington Conference on the Mechanization of Thought Processes*, In (1959)
66. Ernest, D.: *Representations of Commonsense Knowledge*. Morgan Kaufmann, San Francisco (1990)
67. Lenat, D., Guha, R.: *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Addison-Wesley, Boston (1989)
68. Fellbaum, C.: *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, Cambridge (1998)
69. Mueller, E.: *Natural Language Processing with ThoughtTreasure*. Signifonn, New York (1998)
70. Minsky, M.: Commonsense-based interfaces. *Commun. ACM* **43**(8), 67–73 (2000)
71. Singh, P.: The open mind common sense project. *KurzweilAI.net* (2002).
72. Stork, D.: The open mind initiative. *IEEE Intell. Syst.* **14**(3), 16–20 (1999)
73. Chklovski, T.: Learner: a system for acquiring commonsense knowledge by analogy. In: *K-CAP* (2003).
74. Speer, R., Havasi, C., Lieberman, H.: Analogyspace: reducing the dimensionality of common sense knowledge. *AAAI*, In (2008)
75. Havasi, C., Speer, R., Pustejovsky, J., Lieberman, H.: Digital intuition: applying common sense using dimensionality reduction. *IEEE Intell. Syst.* **24**(4), 24–35 (2009)



Sentic Computing

Techniques, Tools, and Applications

Cambria, E.; Hussain, A.

2012, XVIII, 153 p. 39 illus., 35 illus. in color., Softcover

ISBN: 978-94-007-5069-2