Chapter 20
Generating, Refining and Using Sentiment Lexicons

Maarten de Rijke, Valentin Jijkoun, Fons Laan, Wouter Weerkamp, Paul Ackermans, and Gijs Geleijnse

20.1 Introduction

In this chapter, which is based on [7–9], we report on work on the generation, refinement and use of sentiment lexicons that was carried out within the DuOMAn project. The project was focused on the development of language technology to support online *media analysis*. In the area of media analysis, one of the key tasks is collecting detailed information about opinions and attitudes toward specific topics from various sources, both offline (traditional newspapers, archives) and online (news sites, blogs, forums). Specifically, media analysis concerns the following system task: given a topic and list of documents (discussing the topic), find all instances of attitudes toward the topic (e.g., positive/negative sentiments, or, if the topic is an organisation or person, support/criticism of this entity). For every such instance, one should identify the source of the sentiment, the polarity and, possibly, subtopics that this attitude relates to (e.g., specific targets of criticism or support). Subsequently, a (human) media analyst must be able to aggregate the extracted information by source, polarity or subtopics, allowing him to build support/criticism networks etc. [1]. Recent advances in language technology, especially in *sentiment analysis*, promise to (partially) automate this task.

Sentiment analysis is often considered in the context of the following two tasks:

M. de Rijke (✉) · F. Laan · W. Weerkamp
ISLA, University of Amsterdam, Amsterdam, Netherlands
e-mail: derijke@uva.nl; a.c.laan@uva.nl; w.weerkamp@uva.nl

V. Jijkoun
Textkernel BV, Amsterdam, Netherlands
e-mail: jijkoun@textkernel.nl

P. Ackermans · G. Geleijnse
Philips Research Europe, Eindhoven, Netherlands
e-mail: paul.ackermans@philips.com; gijs.geleijnse@philips.com

P. Spyns and J. Odijk (eds.), *Essential Speech and Language Technology for Dutch*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-30910-6_20,

- *Sentiment extraction*: given a set of textual documents, identify phrases, clauses, sentences or entire documents that express attitudes, and determine the polarity of these attitudes [11]; and
- *Sentiment retrieval*: given a topic (and possibly, a list of documents relevant to the topic), identify documents that express attitudes *toward this topic* [21].

How can technology developed for sentiment analysis be applied to media analysis? In order to use a *sentiment extraction* system for a media analysis problem, a system would have to be able to determine which of the extracted sentiments are relevant, i.e., it would not only have to identify targets of extracted sentiments, but also decide which targets are relevant for the topic at hand. This is a difficult task, as the relation between a *topic* (e.g., a movie) and specific targets of sentiments (e.g., acting or special effects in the movie) is not always straightforward, in the face of complex linguistic phenomena such as referential expressions (". . . this beautifully shot *documentary*") or bridging anaphora ("the *director* did an excellent job").

In *sentiment retrieval*, on the other hand, the topic is initially present in the task definition, but it is left to the user to identify sources and targets of sentiments, as systems typically return a list of documents ranked by relevance and opinionatedness. To use a traditional sentiment retrieval system in media analysis, one would still have to manually go through ranked lists of documents returned by the system. To be able to support media analysis, we need to combine the specificity of (phrase- or word-level) sentiment analysis with the topicality provided by sentiment retrieval. Moreover, we should be able to identify sources and specific targets of opinions. Another issue is *evidence* for a system's decision. If the output of a system is to be used to inform actions, the system should present evidence, e.g., highlighting words or phrases that indicate a specific attitude. Most modern approaches to sentiment analysis, however, use various flavors of classification, where decisions (typically) come with confidence scores, but without explicit support.

In the first part of this chapter—Sects. 20.3–20.6—we focus on two of the problems identified above: (1) pinpointing evidence for a system's decisions about the presence of sentiment in text, and (2) identifying specific targets of sentiment. We address these problems by introducing a special type of lexical resource: a topic-specific subjectivity lexicon that indicates specific relevant targets for which sentiments may be expressed; for a given topic, such a lexicon consists of pairs (*syntactic clue*, *target*). We present a method for automatically generating a topic-specific lexicon for a given topic and query-biased set of documents. We evaluate the quality of the lexicon both manually and in the setting of an opinionated blog post retrieval task. We demonstrate that such a lexicon is highly *focused*, allowing one to effectively pinpoint evidence for sentiment, while being competitive with traditional subjectivity lexicons consisting of (a large number of) clue words.

In Sect. 20.7, we address the task of detecting on-topic subjectivity in text. Specifically, we want to (1) tell whether a textual document expresses an attitude (positive or negative) towards a specific topic, and moreover, (2) to find where exactly in the document it is expressed (up to a phrase or at least a sentence). The first task is in the area of *sentiment retrieval*. The simplest approach here consist

of two stages: first, we find texts that are on topic, then we filter out those without attitude [14]. A more elaborate approach is based on the assumption that documents are mixtures of two generative components, one "topical" and one "subjective" [17]. In practice, however, these components are not independent: a word that is neutral w.r.t. one topic can be a good subjectivity clue for another (e.g., compare *hard copy* and *hard problem*). Noticing this, Na et al. [20] generate a topic-specific list of possible clues, based on top relevant documents, and use this list for subjectivity filtering (reranking). In Sects. 20.3–20.6 we argue that such clues are specific not only to the topic, but to the exact target they refer to, e.g., when looking for opinions about a sportsman, *solid* is a good subjectivity clue in the phrase *solid performance* but not in *solid color*.

In Sect. 20.8 we explore the task of experience mining, where the goal is to gain insights into criteria that people formulate to judge or rate a product or its usage. We reveal several features that are likely to prove useful for automatic labeling via classification, over and above lexicon-based opinion spotting.

20.2 Related Work

Much work has been done in sentiment analysis. Here, we discuss work related to Sects. 20.3–20.6 of the chapter in four parts: sentiment analysis in general, domain- and target-specific sentiment analysis, product review mining and sentiment retrieval.

20.2.1 Sentiment Analysis

Sentiment analysis is often seen as two separate steps for determining subjectivity and polarity. Most approaches first try to identify subjective units (documents, sentences), and for each of these determine whether it is positive or negative. Kim and Hovy [11] select candidate sentiment sentences and use word-based sentiment classifiers to classify unseen words into a negative or positive class. First, the lexicon is constructed from WordNet: from several seed words, the structure of WordNet is used to expand this seed to a full lexicon. Next, this lexicon is used to measure the distance between unseen words and words in the positive and negative classes. Based on word sentiments, a decision is made at the sentence level. A similar approach is taken by Wilson et al. [30]: a classifier is learnt that distinguishes between polar and neutral sentences, based on a prior polarity lexicon and an annotated corpus. Among the features used are syntactic features. After this initial step, the sentiment sentences are classified as negative or positive; again, a prior polarity lexicon and syntactic features are used. The authors later explored the difference between prior and contextual polarity [31]: words that lose polarity in context, or whose polarity is reversed because of context. Riloff and Wiebe [24] describe

a bootstrapping method to learn subjective extraction patterns that match specific syntactic templates, using a high-precision sentence-level subjectivity classifier and a large unannotated corpus. In our method, we bootstrap from a subjectivity lexicon rather than a classifier, and perform a topic-specific analysis, learning indicators of subjectivity toward a specific topic.

20.2.2 Domain- and Target-Specific Sentiment

The way authors express their attitudes varies with the domain: An unpredictable movie can be positive, but unpredictable politicians are usually something negative. Since it is unrealistic to construct sentiment lexicons, or manually annotate text for learning, for every imaginable domain or topic, automatic methods have been developed. Godbole et al. [6] aim at measuring overall subjectivity or polarity towards a certain entity; they identify sentiments using domain-specific lexicons. The lexicons are generated from manually selected seeds for a broad domain such as *Health* or *Business*, following an approach similar to [11, 12]. All named entities in a sentence containing a clue from a lexicon are considered targets of sentiment for counting. Choi et al. [4] advocate a joint topic-sentiment analysis. They identify "sentiment topics," noun phrases assumed to be linked to a sentiment clue in the same expression. They address two tasks: identifying sentiment clues, and classifying sentences into positive, negative, or neutral. They start by selecting initial clues from SentiWordNet, based on sentences with known polarity. Next, the sentiment topics are identified, and based on these sentiment topics and the current list of clues, new potential clues are extracted. The clues can be used to classify sentences. Fahrni and Klenner [5] identify potential targets in a given domain, and create a target-specific polarity adjective lexicon. They find targets using Wikipedia, and associated adjectives. Next, the target-specific polarity of adjectives is determined using Hearst-like patterns. Kanayama and Nasukawa [10] introduce polar atoms: minimal human-understandable syntactic structures that specify polarity of clauses. The goal is to learn new domain-specific polar atoms, but these are not target-specific. They use manually-created syntactic patterns to identify atoms and coherency to determine polarity. In contrast to much of the work in the literature, we need to specialise subjectivity lexicons not for a domain and target, but for "topics."

20.2.3 Product Features and Opinions

Much work has been done on the task of mining product reviews, where the goal is to identify features of specific products (such as *picture*, *zoom*, *size*, *weight* for digital cameras) and opinions about these specific features in user reviews. Liu et al. [15] describe a system that identifies such features via rules learned from a manually

annotated corpus of reviews; opinions on features are extracted from the structure of reviews (which explicitly separate positive and negative opinions). Popescu and Etzioni [23] present a method that identifies product features for using corpus statistics, WordNet relations and morphological cues. Opinions about the features are extracted using a hand-crafted set of syntactic rules. Targets extracted in our method for a topic are similar to features extracted in review mining for products. Topics in our setting go beyond concrete products; the diversity and generality of possible topics makes it difficult to apply such supervised or thesaurus-based methods to identify opinion targets. Moreover, we directly use associations between targets and opinions to extract both.

20.2.4 Sentiment Retrieval

At TREC, the Text REtrieval Conference, there has been interest in a specific type of sentiment analysis: opinion retrieval. This interest materialised in 2006 [21], with the opinionated blog post retrieval task. Finding blog posts that are not just about a topic, but also contain an opinion on the topic, proves to be a difficult task [27,28]. Performance on the opinion-finding task is dominated by performance on the underlying document retrieval task (the topical baseline). Opinion finding is often approached as a two-stage problem: (1) identify documents relevant to the query, (2) identify opinions. In stage (2) one commonly uses either a binary classifier to distinguish between opinionated and non-opinionated documents or applies reranking of the initial result list using some opinion score. Opinion add-ons show only slight improvements over relevance-only baselines. The best performing opinion finding system at TREC 2008 is a two-stage approach using reranking in stage (2) [14]. The authors use SentiWordNet and a corpus-derived lexicon to construct an opinion score for each post in an initial ranking of blog posts. This score is combined with the relevance score, and posts are reranked according to this new score. We detail this approach in Sect. 20.6. Later, the authors use domain-specific opinion indicators [20], like "interesting story" (movie review), and "light" (notebook review). This domain-specific lexicon is constructed using feedback-style learning: retrieve an initial list of documents and use the top documents as training data to learn an opinion lexicon. Opinion scores per document are then computed as an average of opinion scores over all its words. Results show slight improvements (+3 %) on mean average precision.

20.3 Generating Topic-Specific Lexicons

In this section we describe how we generate a lexicon of subjectivity clues and targets for a given *topic* and a list of *relevant documents* (e.g., retrieved by a search engine for the topic). As an additional resource, we use a large background corpus

Table 20.1 Examples of subjective syntactic contexts of clue words (based on Stanford dependencies)

Clue word	Syntactic context	Target	Example
To like	Has direct object	*u2*	*I do still like U2 very much*
To like	Has clausal complement	*Criticize*	*I don't like to criticize our intelligence services*
To like	Has *about*-modifier	*Olympics*	*That's what I like about Winter Olympics*
Terrible	Is adjectival modifier of	*Idea*	*It's a terrible idea to recall judges for...*
Terrible	Has nominal subject	*Shirt*	*And Neil, that shirt is terrible!*
Terrible	Has clausal complement	*Can*	*It is terrible that a small group of extremists can ...*

of text documents of a similar style but with diverse subjects; we assume that the relevant documents are part of this corpus as well. As the background corpus, we used the set of documents from the assessment pools of TREC 2006–2008 opinion retrieval tasks (described in detail in Sect. 20.4). We use the Stanford lexicalised parser[1] to extract labeled dependency triples (*head, label, modifier*). In the extracted triples, all words indicate their category (*noun, adjective, verb, adverb,* etc.) and are normalised to lemmas. Figure 20.1 provides an overview of our method; below we describe it in more detail.

20.3.1 Step 1: Extracting Syntactic Contexts

We start with a general domain-independent prior polarity lexicon of 8,821 clue words [30]. First, we identify *syntactic contexts* in which specific clue words can be used to express attitude: we try to find how a clue word can be syntactically linked to targets of sentiments. We take a simple definition of the syntactic context: a single labeled directed dependency relation. For every clue word, we extract all syntactic contexts, i.e., all dependencies, in which the word is involved (as head or as modifier) in the background corpus, along with their endpoints. Table 20.1 shows examples of clue words and contexts that indicate sentiments. For every clue, we only select those contexts that exhibit a high entropy among the lemmas at the other endpoint of the dependencies.

Our entropy-driven selection of syntactic contexts of a clue word is based on the following assumption:

Assumption 1. In text, targets of sentiments are more diverse than sources of sentiments or other accompanying attributes such as location, time, manner, etc. Therefore targets exhibit higher entropy than other attributes.

[1] http://nlp.stanford.edu/software/lex-parser.shtml

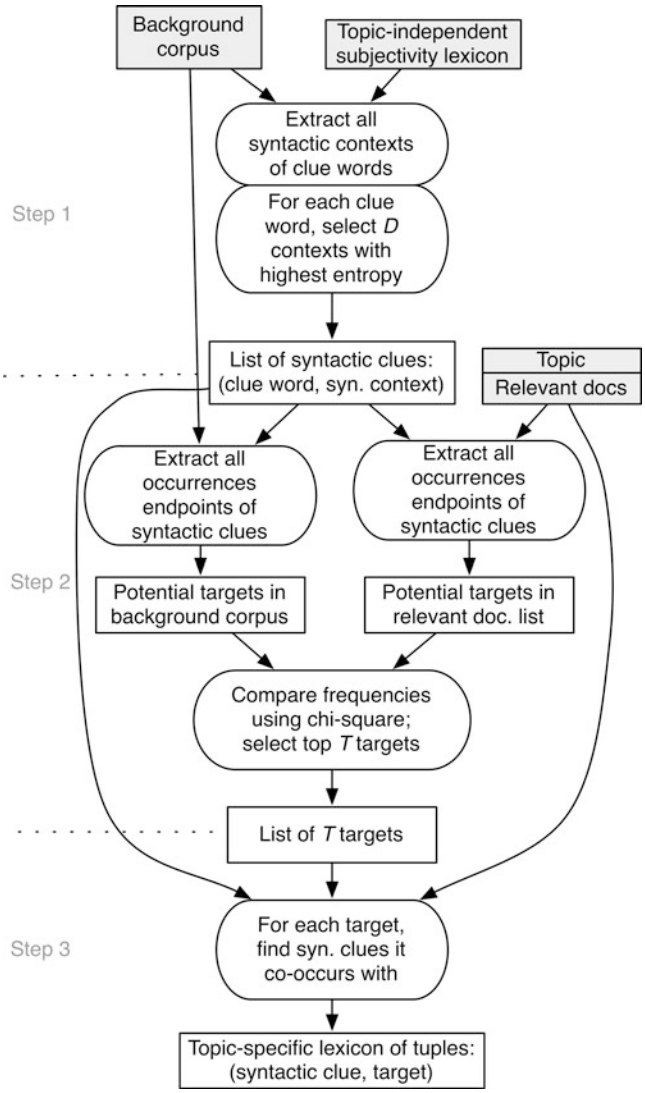

Fig. 20.1 Our method for learning a topic-dependent subjectivity lexicon

For every clue word, we select the top D syntactic contexts whose entropy is at least half of the maximum entropy for this clue. To summarise, at the end of Step 1 of our method, we have extracted a list of pairs (*clue word, syntactic context*) such that for occurrences of the clue word, the words at the endpoint of the syntactic dependency are likely to be targets of sentiments. We call such a pair a *syntactic clue*.

Table 20.2 Examples of targets extracted at Step 2

Topic "Relationship between Abramoff and Bush"
abramoff lobbyist scandal fundraiser bush fund-raiser republican prosecutor tribe swirl corrupt corruption norquist democrat lobbying investigation scanlon reid lawmaker dealings president
Topic "MacBook Pro"
macbook laptop powerbook connector mac processor notebook fw800 spec firewire imac pro machine apple powerbooks ibook ghz g4 ata binary keynote drive modem
Topic: "Super Bowl ads"
ad bowl commercial fridge caveman xl endorsement advertising spot advertiser game super essential celebrity payoff marketing publicity brand advertise watch viewer tv football venue

20.3.2 Step 2: Selecting Potential Targets

Here, we use the extracted syntactic clues to identify words that are likely to serve as specific targets for opinions about the topic in the relevant documents. In this work we only consider individual words as potential targets and leave exploring other options (e.g., NPs and VPs as targets) for future work. In extracting targets, we rely on the following assumption:

Assumption 2. The list of relevant documents contains a substantial number of documents on the topic which, moreover, contain sentiments about the topic.

We extract all endpoints of all occurrences of the syntactic clues in the relevant documents, as well as in the background corpus. To identify potential attitude targets in the relevant documents, we compare their frequency in the relevant documents to the frequency in the background corpus using the standard χ^2 statistics. This technique is based on the following assumption:

Assumption 3. Sentiment targets related to the topic occur more often in subjective context in the set of relevant documents, than in the background corpus. While the background corpus contains sentiments towards very diverse subjects, the relevant documents tend to express attitudes related to the topic.

For every potential target, we compute the χ^2-score and select the top T highest scoring targets. As the result of Steps 1 and 2, as candidate targets for a given topic, we only select words that occur in subjective contexts, and that do so more often than we would normally expect. Table 20.2 shows examples of extracted targets for three TREC topics (see below for a description of our experimental data).

20.3.3 Step 3: Generating Topic-Specific Lexicons

In the last step of the method, we combine clues and targets. For each target identified in Step 2, we take all syntactic clues extracted in Step 1 that co-occur with the target in the relevant documents. The resulting list of triples (*clue word, syntactic context, target*) constitute the lexicon. We conjecture that an occurrence of

a lexicon entry in a text indicates, with reasonable confidence, a subjective attitude towards the target.

20.4 Data and Experimental Setup

We consider two types of evaluation. In the next section, we examine the quality of the lexicons we generate. After that we evaluate lexicons quantitatively using the TREC Blog track benchmark. We apply our lexicon generation method to a collection of documents containing opinionated utterances: blog posts [16]. We perform two preprocessing steps [27, 28]: (1) when extracting plain text from HTML, we only keep block-level elements longer than 15 words (to remove boilerplate material), and (2) we remove non-English posts using TextCat[2] for language detection. We index the collection using Indri,[3] version 2.10 [13]. TREC 2006–2008 came with the task of *opinionated blog post retrieval* [21]. For each year a set of 50 topics was created, giving us 150 topics in total. Every topic comes with a set of relevance judgments: Given a topic, a blog post can be either (1) nonrelevant, (2) relevant, but not opinionated, or (3) relevant and opinionated. TREC topics consist of three fields (*title*, *description*, and *narrative*), of which we only use the *title* field: a query of 1–3 keywords.

We use standard evaluation measures for opinion retrieval: MAP (mean average precision), R-precision (precision within the top R retrieved documents, where R is the number of known relevant documents in the collection), MRR (mean reciprocal rank), P@10 and P@100 (precision within the top 10 and 100 retrieved documents). In the context of media analysis, recall-oriented measures such as MAP and R-precision are more meaningful than early precision-oriented measures. For the opinion retrieval task a document is considered relevant if it is on topic and contains opinions or sentiments towards the topic. We test for significant differences using a two-tailed paired t-test, and report on significant differences for $\alpha = 0.01$ ($^{\blacktriangle}$ and $^{\blacktriangledown}$), and $\alpha = 0.05$ ($^{\triangle}$ and $^{\triangledown}$). For the quantitative experiments in Sect. 20.6 we need a topical baseline: a set of blog posts potentially relevant to each topic. For this, we use the Indri retrieval engine, and apply the Markov Random Fields to model term dependencies in the query [19] to improve topical retrieval. We retrieve the top 1,000 posts for each query.

20.5 Qualitative Analysis of Lexicons

Lexicon size (the number of entries) and selectivity (how often entries match in text) of the generated lexicons vary depending on the parameters D and T introduced above. The two rightmost columns of Table 20.4 show the lexicon size

[2] http://odur.let.rug.nl/~vannoord/TextCat/

[3] http://www.lemurproject.org/indri/

Table 20.3 Posts with highlighted targets (*bold*) and subjectivity clues (*blue*) using topic-independent (*left*) and topic-specific (*right*) lexicons

There are some *tragic* moments like eggs freezing , and predators *snatching* the females and *little* ones-you know the whole *NATURE* thing . . . but this movie is *awesome*	There are some tragic moments l ike eggs freezing , and predators snatching the females and little ones-you know the whole NATURE thing . . . but this **movie** is *awesome*
Saturday was more errands, then spent the evening with Dad and Stepmum, and *finally* was *able* to see March of the Penguins, which was *wonderful*. Christmas Day was *lovely*, surrounded by family, *good* food and drink, and *little* L to play with	Saturday was more errands, then spent the evening with Dad and Stepmum, and finally was able to see March of the **Penguins**, which was *wonderful*. Christmas Day was lovely, surrounded by family, good food and drink, and little L to play with

and the average number of matches per topic. Because our topic-specific lexicons consist of triples (*clue word, syntactic context, target*), they actually contain more words than topic-independent lexicons of the same size, but topic-specific entries are more selective, which makes the lexicon more focused. Table 20.3 compares the application of topic-independent and topic-specific lexicons to on-topic blog text. We manually performed an explorative error analysis on a small number of documents, annotated using the smallest lexicon in Table 20.4 for the topic "March of the Penguins." We assigned 186 matches of lexicon entries in 30 documents into four classes: REL: sentiment towards a relevant target; CONTEXT: sentiment towards a target that is irrelevant to the topic due to context (e.g., opinion about a target "film", but referring to a film different from the topic); IRREL: sentiment towards irrelevant target (e.g., "game" for a topic about a movie); NOSENT: no sentiment at all. In total only 8 % of matches were manually classified as REL, with 62 % classified as NOSENT, 23 % as CONTEXT, and 6 % as IRREL. Among documents assessed as opinionated by TREC assessors, only 13 % did not contain matches of the lexicon entries, compared to 27 % of non-opinionated documents, which does indicate that our lexicon does attempt to separate non-opinionated documents from opinionated.

20.6 Quantitative Evaluation of Lexicons

In this section we assess the quality of the generated topic-specific lexicons numerically and extrinsically. To this end we deploy our lexicons to the task of opinionated blog post retrieval [21]. A commonly used approach to this task works in two stages: (1) identify topically relevant blog posts, and (2) classify these posts as being opinionated or not. In stage 2 the standard approach is to rerank the results from stage 1. We take this approach, as it has shown good performance in the past TREC editions [21] and is fairly straightforward to implement. Our experiments have two goals: to compare the use of topic-independent and topic-specific lexicons for the

Table 20.4 Evaluation of topic-specific lexicons applied to the opinion retrieval task, compared to the topic-independent lexicon. The two rightmost columns show the number of lexicon entries (avg. per topic) and the number of matches of lexicon entries in blog posts (avg. for top 1,000 posts)

Lexicon			MAP	R-prec	MRR	P@10	P@100	|Lexicon|	Hits per doc
No reranking			0.2966	0.3556	0.6750	0.4820	0.3666	–	–
Topic-independent			0.3182	0.3776	**0.7714**	**0.5607**	0.3980	8,221	36.17
D	T	S_{op}							
3	50	count	0.3191	0.3769	0.7276$^{\triangledown}$	0.5547	0.3963	2,327	5.02
3	100	count	0.3191	0.3777	0.7416	0.5573	0.3971	3,977	8.58
5	50	count	0.3178	0.3775	0.7246$^{\triangledown}$	0.5560	0.3931	2,784	5.73
5	100	count	0.3178	0.3784	0.7316$^{\triangledown}$	0.5513	0.3961	4,910	10.06
All	50	count	0.3167	0.3753	0.7264$^{\triangledown}$	0.5520	0.3957	4,505	9.34
All	100	count	0.3146	0.3761	0.7283$^{\triangledown}$	0.5347$^{\triangledown}$	0.3955	8,217	16.72
All	50	okapi	0.3129	0.3713	0.7247$^{\blacktriangledown}$	0.5333$^{\triangledown}$	0.3833$^{\triangledown}$	4,505	9.34
All	100	okapi	0.3189	0.3755	0.7162$^{\blacktriangledown}$	0.5473	0.3921	8,217	16.72
All	200	okapi	**0.3229**$^{\blacktriangle}$	**0.3803**	0.7389	0.5547	**0.3987**	14,581	29.14

opinionated post retrieval task, and to examine how settings for the parameters of the lexicon generation affect the empirical quality.

To rerank a list of posts retrieved for a given topic, we opt to use the method that showed best performance at TREC 2008. The approach taken by Lee et al. [14] linearly combines a (topical) relevance score with an opinion score for each post. For the opinion score, terms from a (topic-independent) lexicon are matched against the post content, and weighted with the probability of term's subjectivity. Finally, the sum is normalised using the Okapi BM25 framework. The final opinion score S_{op} is computed as in Eq. 20.1:

$$S_{op}(D) = \frac{Opinion(D) \cdot (k_1 + 1)}{Opinion(D) + k_1 \cdot (1 - b + \frac{b \cdot |D|}{avgdl})}, \tag{20.1}$$

where k_1, and b are Okapi parameters (set to their default values $k_1 = 2.0$, and $b = 0.75$), $|D|$ is the length of document D, and $avgdl$ is the average document length in the collection. The opinion score $Opinion(D)$ is calculated as $Opinion(D) = \sum_{w \in O} P(sub|w) \cdot n(w, D)$, where O is the set of terms in the sentiment lexicon, $P(sub|w)$ indicates the probability of term w being subjective, and $n(w, D)$ is the number of times term w occurs in document D. The opinion scoring can weigh lexicon terms differently, using $P(sub|w)$; it normalises scores to cancel out the effect of varying document sizes. We use the method described above, and plug in the MPQA polarity lexicon.[4] We compare the results of using this topic-independent lexicon to the topic-dependent lexicons our method generates,

[4] http://www.cs.pitt.edu/mpqa/

which are also plugged into the reranking of [14]. In addition to using Okapi BM25 for opinion scoring, we also consider a simpler method. As we observed in Sect. 20.5, our topic-specific lexicons are more selective than the topic-independent lexicon, and a simple number of lexicon matches can give a good indication of opinionatedness of a document: $S_{op}(D) = \min(n(O, D), 10)/10$, where $n(O, D)$ is the number of matches of the term of sentiment lexicon O in document D.

There are several parameters that we can vary when generating a topic-specific lexicon and when using it for reranking: D: the number of syntactic contexts per clue; T: the number of extracted targets; $S_{op}(D)$: the opinion scoring function; and α: the weight of the opinion score in the linear combination with the relevance score. Note that α does not affect the lexicon creation, but only how the lexicon is used in reranking. Since we want to assess the quality of lexicons, we factor out α by selecting the best setting for each lexicon (including the topic-independent) and each evaluation measure.

In Table 20.4 we present the results of evaluation of several lexicons in the context of opinionated blog post retrieval. First, we note that reranking using all lexicons significantly improves over the relevance-only baseline for all evaluation measures. When comparing topic-specific lexicons to the topic-independent one, most of the differences are not statistically significant, which is surprising given the fact that most topic-specific lexicons we evaluated are substantially smaller (see the two rightmost columns in the table). The smallest lexicon in Table 20.4 is seven times more selective than the general one, in terms of the number of lexicon matches per document. The only measure where the topic-independent lexicon consistently outperforms topic-specific ones, is MRR, which depends on a single relevant opinionated document high in a ranking. The general lexicon easily finds a "obviously subjective" posts (those with heavily used subjective words), but is not better at detecting less obvious ones, as indicated by the recall-oriented MAP and R-precision. Increasing the number of syntactic contexts considered for a clue word (parameter D) and the number of selected targets (parameter T) leads to substantially larger lexicons, but only gives marginal improvements when lexicons are used for opinion retrieval. This shows that our bootstrapping method is effective at filtering out non-relevant sentiment targets and syntactic clues. The choice of opinion scoring function (Okapi or raw counts) depends on the lexicon size: for smaller, more focused lexicons unnormalised counts are more effective; simple presence of a sentiment clue in text is a good indication of subjectivity. For larger lexicons an overall subjectivity scoring of texts has to be used, which can be hard to interpret for (media analysis) users.

20.7 Bootstrapping Subjectivity Detection

In Sects. 20.3–20.6 we have described a method for learning pairs (clue, target) for a given topic in an unsupervised manner, using syntactic dependencies between clues and targets. We go beyond the subjectivity lexicon generation methods from

Sects. 20.3–20.6, with the goal of improving subjectivity spotting. We directly evaluate the performance on the task of detecting on-topic subjectivity at the sentence level, not on sentiment retrieval with entire documents. Our method does not use a seed set.

20.7.1 Method

We start with a topic T (a textual description) and a set $R = \{d_1, \ldots, d_N\}$ of documents deemed relevant to T. The method uses a general-purpose list of subjectivity clues L (in our experiments, the well-known MPQA lexicon [29]). We use a background corpus BG of documents of a similar genre, covering many topics beside T. We use the Stanford syntactic parser to extract dependency relations in all sentences in all documents. Our method outputs a set of triples $\{(c_i, r_i, t_i)\}$, where c_i is a subjective clue, t_i a subjectivity target and r_i a dependency relation between the two words. We interpret an occurrence of such a triple as an indication of sentiment relevant to T, specifically directed at t_i.

We assume that a given topic can be associated with a number of related targets (e.g., opinions about a sportsman may cover such targets as *performance, reaction, serve,* etc.) and each target has a number of possible clues expressing attitude towards it (e.g., *solid performance*). We assume that clues and targets are typically syntactically related (e.g., the target *serve* can be a direct object of clue *to like*), and every clue has syntactic relations connecting it to possible targets (e.g., for *to like* only the direct object can be a target, but not the subject, a adverbial modifier, etc.).

20.7.1.1 Step 1: Initial Clue Scoring

For every possible clue $c \in L$ and every type of syntactic relation r that can originate from it in the background corpus, we compute a *clue score* $s_{clue}(c, r)$ as the entropy of words at the other endpoint of r in BG (normalised between 0 and 1 for all c and r). The clue score gives an initial estimate of how well (c, r) may work as a subjectivity clue. Here, we follow the intuition of Sects. 20.3–20.6: targets are more diverse than other syntactic neighbors of clues.

20.7.1.2 Step 2: Target Scoring

For every word $t \in R$ we determine its target score that tells us how likely t is an opinion target related to topic T. Targets are words that occur unusually often in subjective contexts in relevant documents. First, we compute $C_R(t) = \sum s_{clue}(c, r)$ for all occurrences of the syntactic relation r between words c and t in corpus R. Similarly, we compute $C_{BG}(t)$ for the background corpus BG. We view $C_R(\cdot)$ and $C_{BG}(\cdot)$ as (weighted) counts, and compute a parsimonious language model $p_R(\cdot)$

Table 20.5 Test results on a sentence classification task

Method				P	R	F_1
Method of Sects. 20.3–20.6				0.23	0.31	0.26
R	K	N	M			
$r + 100$	4	10	50	0.42	0.13	0.20
$r + 100$	4	20	50	0.45	0.17	0.25
$r + 100$	4	30	50	0.35	0.26	0.28
$r + 100$	4	40	50	0.32	0.29	0.30
$r + 100$	4	50	50	0.20	0.30	0.24
$r + 100$	4	60	50	0.19	0.32	0.24
$r + 100$	4	70	50	0.14	0.35	0.20
$r + 100$	4	40	30	0.32	0.21	0.25
$r + 100$	4	40	40	0.32	0.23	0.27
$r + 100$	4	40	50	0.32	0.29	0.30
$r + 100$	4	40	60	0.30	0.29	0.29
$r + 100$	4	40	70	0.29	0.30	0.29
$r + 100$	4	40	50	0.32	0.29	0.30
100	4	40	50	0.27	0.22	0.24
r	4	40	50	0.21	0.17	0.19

using a simple EM algorithm [18]. We also compute a language model $p_B G(\cdot)$ from counts $C_{BG}(\cdot)$ by simple normalisation. Finally, we define the target score of a word t as the likelihood that the occurrence of t in R comes from $p_R(\cdot)$ rather than $p_B G(\cdot)$:

$$s_{tgt}(t) = \frac{\gamma \cdot p_{tgt}(t)}{\gamma \cdot p_{tgt}(t) + (1 - \gamma) \cdot p_{BG}(t)}.$$

20.7.1.3 Step 3: Clue Scoring

Mirroring Step 2, we now use target scores to compute better estimates for clue scores. Here, our intuition is that good subjectivity clues are those that occur unusually often near possible opinion targets for a given topic. The computation is similar to Step 2, with $s_{clue}(c, r)$ and $s_tgt(t)$ interchanged: we compute weighted counts, a parsimonious model and, finally, the updated $s_{clue}(c, r)$. Now, we iterate Step 2 and Step 3, each time updating $s_{tgt}(\cdot)$ and $s_{clue}(\cdot, \cdot)$, respectively, based on the values at the previous iteration. After K iterations we select N targets and M pairs (clue, relation) with the highest scores. We check which of the N targets co-occur with which of the M clues in R.

20.7.2 Experiments and Results

We evaluate different versions of our method on the following sentence classification task: for a given topic and a list of documents relevant to the topic, we

need to identify sentences that express opinions relevant to the topic. We compute precision, recall and F-score for detection of relevant opinionated sentences. We use the NTCIR-6 [25] and NTCIR-7 [26] Opinion Analysis datasets, containing judgements for 45 queries and 12,000 sentences. In order to understand how the quality of relevant documents affects the performance of the method, we selected R to be (1) R_{100}: top 100 document retrieved from the NTCIR-6/7 English collection using Lucene, (2) R_r: only documents with at least one relevant (not necessarily opinionated) sentence as identified by NTCIR annotators, and (3) R_{r+100} the union of (1) and (3). We ran the method with different numbers of iterations (K), selected targets (N) Table 20.5 shows the results, the overall performance stabilises at $K \leq 5$. The table included above shows the evaluation results. We see that reducing the number of selected targets (N) improves precision but harms recall. Changing the number of selected clues (M) has little effect on precision: since for detecting opinionatedness we combine clues with targets, noise in clues does not necessarily lead to drop in precision. Overall, we notice that in the best setting ($K = 4$, $N = 40$, $M = 50$) the method outperforms the method described in Sects. 20.3–20.6 (significantly, at $p = 0.05$, using t-test). Performance of the method varies substantially per topic (F_1 between 0.13 and 0.48), but the optimal values for parameters are stable for high-performing topics (with $F_1 > 0.26$).

20.8 Mining User Experiences from Online Forums

We change tack again and report on an exploratory study. It touches on an important step after the initial groundwork laid down by lexicon generation and refinement of the type described so far: mining user experiences. Let us provide some background. Recent years have shown a large increase in the usage of content creation platforms aimed at the general public. User generated data contains emotional, opinionated, sentimental, and personal posts. This feature makes it an interesting data source for exploring new types of text analysis, as is shown by research on sentiment analysis [22], opinion retrieval [21], and mood detection [2]. We introduce the task of *experience mining*. Here, the goal is to gain insights into criteria that people formulate to judge or rate a product or its usage. We focus on reports of experiences with products.

20.8.1 Motivation

Our main use-case is user-centered design for product development. User-centered design [3] is an innovation paradigm where users of a product are involved in each step of the research and development process. The first stage of the product design process is to identify unmet needs and demands of users for a specific product or a class of products. Although statements found in such platforms may not always be representative for the general user group, they can accelerate user-centered design.

20.8.2 Experience Mining

Experiences are particular instances of personally encountering or undergoing something. We want to identify experiences about a specific *target product*, that are *personal*, involve an *activity* related to the target and, moreover, are accompanied by *judgements or evaluative statements*. Experience mining is related to sentiment analysis and opinion retrieval, in that it involves identifying attitudes; the key difference is, however, that we are looking for *attitudes towards specific experiences* with products, not attitudes towards the products themselves.

20.8.3 An Explorative Study

To assess the feasibility of automatic experience mining, we carried out an explorative study: we asked human assessors to find experiences in actual forum data and then examined linguistic features likely to be useful for identifying experiences automatically. We acquired data by crawling two forums on shaving,[5] with 111,268 posts written by 2,880 users. Two assessors searched for posts on five specific target products using a standard keyword search, and labeled each result post as: (1) reporting no experience, or (2) reporting an off-target experience, or (3) reporting an on-target experience. Posts should be marked as reporting an experience only if (1) the author explicitly reports his or someone else's (a concrete person's) use of a product; and (2) the author makes some conclusions/judgements about the experience. In total, 203 posts were labeled, with 101 posts marked as reporting an experience by at least one assessor (71 % of those an on-target experience). The inter-annotator agreement was 0.84, with Cohen's $\kappa = 0.71$. If we merge on- and off-target experience labels, the agreement is 0.88, with $\kappa = 0.76$. The high level of agreement demonstrates the validity of the task definition. We considered a number of linguistic features and compared posts reporting experience (on- or off-target) to the posts with no experience. Table 20.6 lists the features and the comparison results. The subjectivity score is lower for experience posts: our task is indeed different from sentiment retrieval! Experience posts are on average twice as long as non-experience posts and contain more sentences with pronoun *I*. They also contain more content (non-modal) verbs, especially past tense verbs. Table 20.7 presents an analysis of the verb use. Experience posts contain more verbs referring to concrete actions than to attitude and perception. It remains to be seen whether this observation can be quantified using resources such as standard semantic verb classification (*state, process, action*), WordNet verb hierarchy or FrameNet semantic frames.

[5] http://www.shavemyface.com and http://www.menessentials.com/community.

Table 20.6 Comparison of surface features; $p(\cdot)$ denotes probability

	Mean and deviation in posts with/without experience	
Feature	*With*	*Without*
Subjectivity score[6]	0.07 ± 0.23	0.17 ± 0.35
Polarity score[6]	0.87 ± 0.30	0.77 ± 0.38
#words per post	102.57 ± 80.09	52.46 ± 53.24
#sentences per post	6.00 ± 4.16	3.34 ± 2.33
# words per sentence	17.07 ± 4.69	15.71 ± 7.61
#questions per post	0.32 ± 0.63	0.54 ± 0.89
p(post contains question)	0.25 ± 0.43	0.33 ± 0.47
#*I*'s per post	5.76 ± 4.75	2.09 ± 2.88
#*I*'s per sentence	1.01 ± 0.48	0.54 ± 0.60
p(sentence in post contains *I*)	0.67 ± 0.23	0.40 ± 0.35
#non-modal verbs per post	19.62 ± 15.08	9.82 ± 9.57
#non-modal verbs per sent.	3.30 ± 1.18	2.82 ± 1.37
#modal verbs per sent.	0.22 ± 0.22	0.26 ± 0.36
Fraction of past-tense verbs	0.26 ± 0.17	0.17 ± 0.19
Fraction of present tense verbs	0.42 ± 0.18	0.41 ± 0.23

Table 20.7 Frequent past tense verbs following *I* with relative frequencies

In posts with experience	In posts without experience
Used 0.15, found 0.09, bought 0.07, tried 0.07, got 0.07, went 0.07, started 0.05, switched 0.04, liked 0.03, decided 0.03	Got 0.09, thought 0.09, switched 0.06, meant 0.06, used 0.06, went 0.06, ignored 0.03, quoted 0.03, discovered 0.03, heard 0.03

20.9 Conclusion

We started this chapter by describing a bootstrapping method for deriving a topic-specific lexicon from a general purpose polarity lexicon. We evaluated the quality of the lexicons generated by our method both manually and using a TREC Blog track test set for opinionated blog post retrieval. Although the generated lexicons can be an order of magnitude more selective, they maintain, or even improve, the performance of an opinion retrieval system. In future work, we want to look at more complex syntactic Choi et al. [4] report that many errors are due to exclusive use of unigrams. We also want to extend potential opinion targets to include multi-word phrases (NPs and VPs).

Second, in this chapter we also described a method for automatically generating subjectivity clues for a specific topic and a set of (relevant) document, evaluating it on the task of classification of sentences w.r.t. subjectivity, demonstrating improvements over previous work. Here, we plan to incorporate more complex syntactic

[6]Computed using LingPipe: http://alias-i.com/lingpipe.

patterns in our clues (going beyond word-word relations) and study the effect of user feedback with the view of implementing an interactive system.

Finally, we explored the novel task of experience mining. Users of products share their experiences, and mining these could help define requirements for next-generation products. We developed annotation guidelines for labeling experiences, and used them to annotate data from online forums. An initial exploration revealed multiple features that might prove useful for automatic labeling via classification.

Acknowledgements In addition to funding by the STEVIN programme, this research was also partially supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements nr 258191 (PROMISE Network of Excellence) and 288024 (LiMoSINe project), the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.061.814, 612.061.815, 640.004.802, 380-70-011, 727.011.005, the Center for Creation, Content and Technology (CCCT), the Hyperlocal Service Platform project funded by the Service Innovation & ICT program, the WAHSP project funded by the CLARIN-NL program, under COMMIT project Infiniti and by the ESF Research Network Program ELIAS.

References

1. Altheide, D.: Qualitative Media Analysis. Sage, Thousand Oaks (1996)
2. Balog, K., Mishne, G., de Rijke, M.: Why are they excited?: identifying and explaining spikes in blog mood levels. In: EACL '06, Trento, pp. 207–210 (2006)
3. Buxton, B.: Sketching User Experiences: Getting the Design Right and the Right Design. Morgan Kaufmann, San Francisco (2007)
4. Choi, Y., Kim, Y., Myaeng, S.-H.: Domain-specific sentiment analysis using contextual feature generation. In: TSA '09, pp. 37–44. ACM, New York (2009)
5. Fahrni, A., Klenner, M.: Old wine or warm beer: target-specific sentiment analysis of adjectives. In: AISB 2008 Convention, Aberdeen, pp. 60–63 (2008)
6. Godbole, N., Srinivasaiah, M., Skiena, S.: Large-scale sentiment analysis for news and blogs. In: ICWSM '07, Denver (2007)
7. Jijkoun, V., de Rijke, M.: Bootstrapping subjectivity detection. In: SIGIR '11, Beijing (2011)
8. Jijkoun, V., de Rijke, M., Weerkamp, W.: Generating focused topic-specific sentiment lexicons. In: ACL '10, Uppsala (2010a)
9. Jijkoun, V., de Rijke, M., Weerkamp, W., Ackermans, P., Geleijnse, G.: Mining user experiences from online forums: an exploration. In: NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media, Los Angeles (2010b)
10. Kanayama, H., Nasukawa, T.: Fully automatic lexicon expansion for domain-oriented sentiment analysis. In: EMNLP '06, Sydney, pp. 355–363 (2006)
11. Kim, S., Hovy, E.: Determining the sentiment of opinions. In: COLING 2004, Geneva (2004)
12. Kim, Y., Choi, Y., Myaeng, S.-H.: Generating domain-specific clues using news corpus for sentiment classification. In: ICWSM '10, Washington, DC (2010)
13. Lavrenko, V., Croft, B.: Relevance-based language models. In: SIGIR '01, New Orleans (2001)

14. Lee, Y., Na, S.-H., Kim, J., Nam, S.-H., Jung, H.-Y., Lee, J.-H.: KLE at TREC 2008 blog track: blog post and feed retrieval. In: TREC 2008, Gaithersburg (2008)
15. Liu, B., Hu, M., Cheng, J.: Opinion observer: analyzing and comparing opinions on the web. In: WWW '05, Chiba (2005)
16. Macdonald, C., Ounis, I.: The TREC Blogs06 collection: creating and analysing a blog test collection. Technical Report TR-2006-224, University of Glasgow (2005)
17. Mei, Q., Ling, X., Wondra, M., Su, H., Zhai, C.: Topic sentiment mixture: modeling facets and opinions in weblogs. In: WWW '07, Banff, pp. 171–180 (2007)
18. Meij, E., Weerkamp, W., Balog, K., de Rijke, M.: Parsimonious relevance models. In: SIGIR '08, Singapore (2008)
19. Metzler, D., Croft, W.B.: A markov random feld model for term dependencies. In: SIGIR '05, Salvador, pp. 472–479 (2005)
20. Na, S.-H., Lee, Y., Nam, S.-H., Lee, J.-H.: Improving opinion retrieval based on query-specific sentiment lexicon. In: ECIR '09, Toulouse, pp. 734–738 (2009)
21. Ounis, I., Macdonald, C., de Rijke, M., Mishne, G., Soboroff, I.: Overview of the TREC 2006 blog track. In: TREC 2006, Gaithersburg (2007)
22. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Found. Trends Inf. Retr. **2**, 1–135 (2008)
23. Popescu, A.-M., Etzioni, O.: Extracting product features and opinions from reviews. In: HLT/EMNLP '05, Vancouver (2005)
24. Riloff, E., Wiebe, J.: Learning extraction patterns for subjective expressions. In: EMNLP '03, Sapporo, Japan (2003)
25. Seki, Y., Evans, D.K., Ku, L.-W., Chen, H.-H., Kando, N., Lin, C.-Y.: Overview of opinion analysis pilot task at NTCIR-6. In: NTCIR-6, Tokyo (2007)
26. Seki, Y., Evans, D.K., Ku, L.-W., Sun, L., Chen, H.-H., Kando, N.: Overview of multilingual opinion analysis task at NTCIR-7. In: NTCIR-7, Tokyo (2008)
27. Weerkamp, W., Balog, K., de Rijke, M.: A generative blog post retrieval model that uses query expansion based on external collections. In: ACL-IJCNLP 2009, Singapore (2009)
28. Weerkamp, W., de Rijke, M.: Credibility improves topical blog post retrieval. In: ACL-08: HLT, Columbus, pp. 923–931 (2008)
29. Wiebe, J., Wilson, T., Cardie, C.: Annotating expressions of opinions and emotions in language. Lang. Resour. Eval. **39**, 165–210 (2005)
30. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: HLT '05, Vancouver, Canada, pp. 347–354 (2005)
31. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity: an exploration of features for phrase-level sentiment analysis. Comput. Linguist. **35**(3), 399–433 (2009)