Maximizing Insight from Unstructured Data

Sundar Varadarajan is head of the business intelligence and analytics practice at Hexaware Technologies. sundarv@hexaware.com

Soundarapandian is a senior architect in the business intelligence and analytics center of excellence at Hexaware Technologies. soundrapandiana@hexaware.com

Sundar Varadarajan and Soundarapandian

Abstract

Extracting meaningful information from unstructured data, in particular text data, is a complex process. Although text analytics engines exist that enable extraction with sufficient accuracy, the process must be supplemented with domain- and problem-specific knowledge and enhanced to handle the nuances and peculiarities of text sources.

The approach we describe in this article involves an ontology, taxonomy definitions, and extraction, as well as specific preprocessing steps (including stratified sampling, structuring phrase extraction in multiple layers, and maximizing accuracy and coverage through iteration). The domain- and problem-specific knowledge that is defined as an extension of the core linguistic engine is a key part of the information extraction layer and plays a critical part in achieving accurate results. We have worked with a variety of text sources, such as e-mail exchanges, chats, tweets, and Facebook and YouTube comments.

We present the approach, the framework employed for supplemental domain knowledge, and the enhancements necessary to significantly improve the accuracy of text analysis results. We illustrate all this using specific examples and scenarios we have successfully applied for some of our customers.

Introduction

In our customer scenarios, we have worked with textual data, written primarily in English. For this article, we will limit our discussion to sentiment analysis in several problem domains and scenarios. In most cases, the sources of text are tweets from Twitter, Facebook comments, chat text, or text data captured in CRM systems. Specific problem areas addressed include product comments, services feedback analysis from a CRM system, student chat in an online university, and review comments on a

CORE PHRASE	SENTIMENT EXPRESSED
Training videos	Beneficial (positive sentiment)
Student financial aid program	Not providing enough coverage (negative sentiment)
Auditorium	It's big (neutral sentiment)

Table 1: Examples of core phrases and sentiments.

CATEGORY	SENTIMENT COUNT
Teaching staff	7,455 positive comments and 150 negative comments
Support staff	10,523 positive comments and 189 negative comments

Table 2: Examples of categories and sentiment counts.

INPUT TEXT	CORE PHRASE AND SENTIMENT	VALIDATION OF OUTPUT
The latest version of Product X is fabulous	Core phrase: Product X Sentiment: Positive	Core phrase: Correctly identified Sentiment: Correctly identified
While the courses are very well structured, the fees are not affordable	Core phrase: Courses Sentiment: Negative	Core phrase: Partially correct (second category, "fees," was missed) Sentiment: Incorrectly identified
I do not think it is a bad product	Core phrase: Product Sentiment: Negative	Core phrase: Correct Sentiment: Incorrectly identified

Table 3: An example of core phrases and sentiment validation.

restaurant ratings site. In all cases, we looked for sentiments expressed. Uniformly, we noticed that sentiment analysis engines could not correctly capture the sentiments in a large number of cases—regardless of the text analysis engine and problem domain. The level of accuracy in sentiment capture varied between 50 and 70 percent.

In this article, we assume a text analysis/sentiment analysis engine is already in use. We describe a methodology for improving accuracy that can be applied on top of any text/sentiment analysis engine. Most of these engines use machine translation techniques to identify sentiments as positive, negative, or neutral. Our approach enhances this process with deeper linguistic analysis that is performed using rule-based information extraction techniques to identify the sentiment more accurately and extract the details of the entities about which the sentiment is expressed.

Terms and Definitions

Before beginning our discussion, we need to explain how we will use several terms in this article.

We use the term *core phrases* to describe business-critical phrases for a specific problem scenario. We use the term *sentiments* to mean the feelings expressed about the core phrases (perhaps on a product or service or anything that is of business importance). Sentiments can be positive, negative, or neutral.

Table 1 provides examples of core phrases and sentiments expressed about them.

Categories are the core concepts specific to the domain or problem and are the subjects of the sentiments. The final sentiment analysis consists of categories and sentiments expressed. For example, Table 2 shows the sentiment count extracted from a set of comments for a university.

The Problem with Sentiment Analysis Engines

Although we have tried using sentiment analysis engines out of the box, the results are often unsatisfactory. The accuracy of derived sentiments, and their relevance to our desired context, fell short of expectations. This required that we develop an approach/methodology and apply techniques to improve the accuracy and relevance of sentiment extraction. Our solution primarily used linguistic processing and rule-based information extraction techniques.

Out-of-the-box sentiment analysis engines typically extract specific sentiments, which are usually described

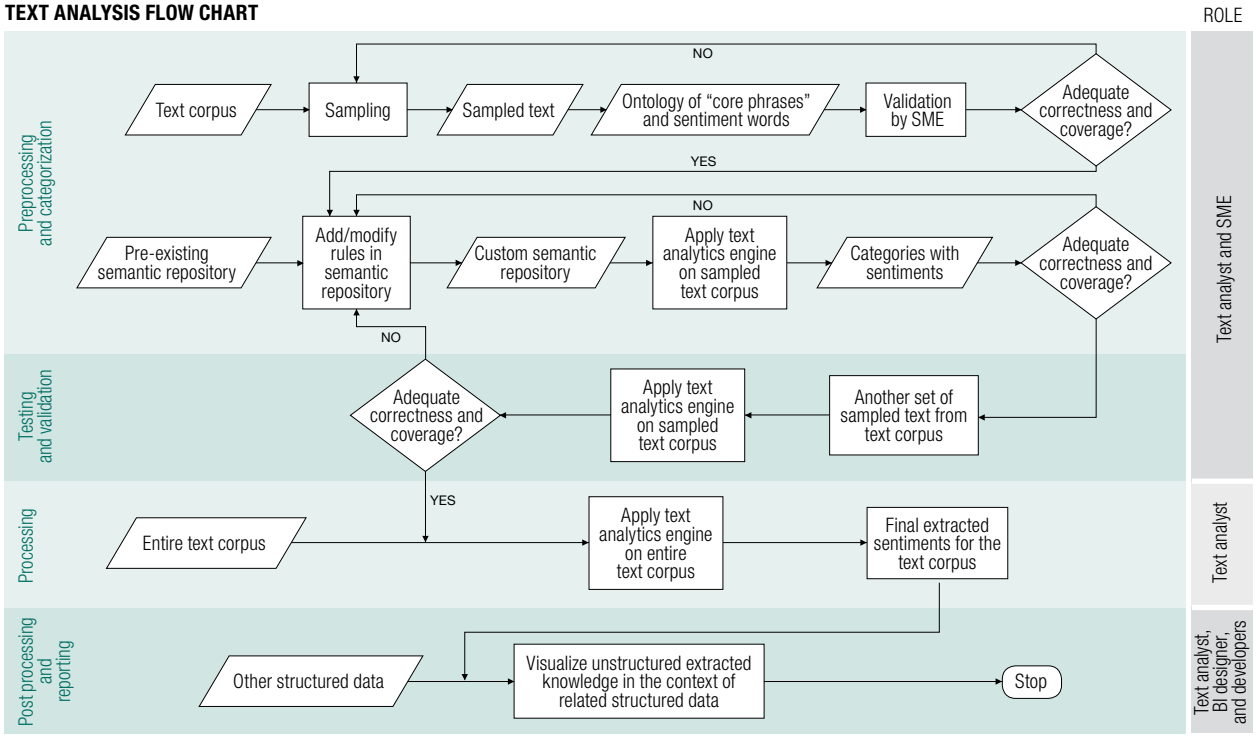


Figure 1: The text analysis process.

as positive, negative, or neutral. They may be ranked on a scale (such as -10 to +10) from "strong negative" to "strong positive" expressions. It is also important to extract the core words about which the sentiment is expressed. See Table 3 for several samples of sentiment extraction.

In many cases, input text contains well-framed sentences with good grammar, but in many other cases we must deal with the problems of underpunctuated text, spelling mistakes, and grammatical errors.

Methodology

Our methodology helps overcome some of the issues and challenges of sentiment analysis.

A key component of our solution and approach is the semantic repository, which must be fine-tuned to deal with domain- and problem-specific nuances. Without this fine-tuned repository, the text analysis output tends to be too generic and suffers from inaccurate results. We will describe the fine-tuning of the semantic repository in more detail as we explain the stages of text analysis.

The flow chart in Figure 1 illustrates how accurate sentiments can be derived from unstructured text sources. We do this in five stages. The chart also indicates the roles of those involved in the different stages.

For typical sentiment extraction and analysis of a scenario or domain (such as student chat analysis for an online university, extracting from 10 to 15 core concepts and sentiments), our experience has been that the preprocessing stage can take between 8 and 10 weeks of effort to fine-tune extraction semantics and rules, and the testing and validation stage can take about 2 weeks. The timeline for establishing the post-processing steps can vary depending on the kinds of visualizations designed and the integration with other structured content for integrated BI information delivery.

Stage 1: Preprocessing

This stage is the longest of all the steps and consumes the majority of the project time. In this stage, our goal is to understand the text source.

Preprocessing starts with collecting unstructured data related to a specific problem from one or more sources. For example, chat session comments are a source of unstructured data for analyzing student opinions about an online university. E-mail messages serve as a second source of unstructured content. The larger the amount of extracted text, the better the accuracy of the analysis. In the case of sentiment analysis from Facebook, we normally extract 4 to 5 million comments. The extracted (and huge) volume of text data is referred to as the "text corpus."

Once the text corpus is in place, we perform a stratified sampling to select a smaller set of comments. We use the initial stratified sample of comments as the basis for extracting categories of business importance as well as for later steps such as sentiment extraction. Using a stratified sample helps to ensure adequate accuracy of subsequent results. We have noted that for 100,000 tweets, a sample size of about 400 is often adequate for a 95 percent confidence level. This representative sample helps us to understand the huge text corpus with minimum effort. (The sample size may need to be increased if we determine that the accuracy level is insufficient during the testing and validation phase.)

From this sample, we manually identify core phrases (and their variations) and sentiment-indicative phrases associated with these core concepts.

Subject matter experts (SMEs) validate the core phrases and sentiment phrases; they repeat the tasks in this stage until there are a sufficient number of core phrases and sentiment-indicators. For example, when analyzing student chat in an online university to understand their sentiments on a variety of topics such as tuition cost, course curriculum, and university infrastructure, an SME is required—one who can look at the different phrases extracted and relate them to the topics of interest, validating the phrases that are meaningful and relevant to the objective.

In the next step, we update the existing semantic repository with the newly identified core phrases and sentiments. We use the revised repository to extract core phrases and sentiments from the stratified text samples.

From the samples, the engine extracts:

1. Core phrases and variations
2. Sentiments related to core phrases and variations
3. Other phrases (related to concepts of interest) and variations
4. Sentiment phrases related to these "other phrases" and variations
5. Other phrases without any associated sentiment

The first and second items in this list are the information of interest for analysis. We manually evaluate the other items to determine whether we should add them to the core phrases. When variations and additions to the core phrases are found, they can help fine-tune and further enrich the semantic repository.

Stage 2: Categorization

Categorization is a part of the preprocessing stage. We use the following three categorization methods:

1. Categorization Based on Context

Typically, semantic repositories associated with text analytics engines contain certain basic core phrases and concepts, as well as sentiment-indicative phrases, all of which are configured for common usage scenarios. Sometimes, certain phrases or concepts are specific to the context of a given domain and may warrant special treatment.

For example, if we are dealing with comments about the customer service of a health and hygiene company, the phrase "waste management" refers to a service provided by the company. However, many sentiment analysis engines extract a negative sentiment when they encounter the word "waste" in any general textual comment. In the context of this specific problem domain, this word must be treated differently. The comments may be capturing a positive sentiment about the "waste management" services of the company.

We address this by overriding the existing generic repository with rules specific to the problem domain and ensuring that "waste" in this context is not treated as a negative sentiment, but instead as relating to a service.

Once the SME identifies "waste management" as a service and core phrase of interest, the text analyst includes this core phrase in the custom repository and adds the specialized rule.

2. Categorization Based on Linguistic Variations

There are many linguistic aspects to consider when processing text. Synonyms and specific thesauri are used to handle phrases that are to be placed in the same category. For example, the category "faculty" can be indicated by "teacher," "professor," "instructor," "lecturer," "mentor," and so on, by some variations in the number (singular or plural), or by specific references to people's names.

Similarly, different verb forms and different usage of active or passive voice, if not already considered in the underlying linguistic engine, should be handled with customized semantic repository rules.

3. Categorization Based on Specific Instances

Specific proper noun phrases in a particular domain (for example, airlines-related comments) may describe a core concept. For example, the terms A777, A380, and A787 can be categorized together as "aircraft."

SMEs validate the output of such categorization, a process that typically involves a few iterations.

A Closer Look: The Semantic Repository

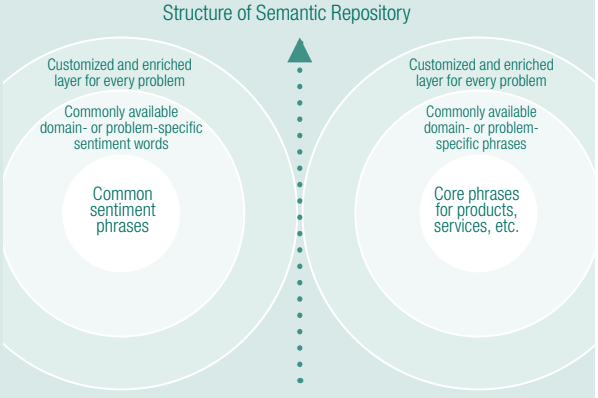
Any text analysis engine uses a semantic repository that captures core phrases used to extract basic concepts and sentiments expressed about them. We have found that this semantic repository needs to be enhanced and customized and must be made specific to the problem and domain we are working with.

Here is an example of a typical collection in a semantic repository for a university:

BASIC COLLECTION	DOMAIN- OR PROBLEM-SPECIFIC
Cities	Fees/cost
Countries	Faculty
States	Environment/facilities
ZIP codes	Curriculum
Products	Tuition fee
Opinions	Financial support
Sentiments	Educational loan
Emoticons	

Text analysis engines provide ways and means to customize or extend existing repositories or help users create new ones.

The semantic repository structure is shown in the following diagram:



We have found this layered structure immensely helpful, especially when dealing with solutions for various problems across various domains or industries. It applies to both core phrases and related sentiment phrases extraction.

The structure improves reusability and extensibility when deploying solutions across several problem domains. It is possible to further expand the number of layers to more granular and extensive domains, if required. Having these specific custom layers on top of a generic layer (which is typically provided out of the box in text analytics tools) is a key contributor to fine-tuning the accuracy of text analytics and sentiment extraction.

CATEGORIES FOR CORE PHRASES	SENTIMENT	SENTIMENT COUNT
AAA Online University	Positive	58,723
Tuition fee	Negative	58,324
Support staff	Negative	47,232
Teaching	Positive	46,232
AAA Online University	Negative	8,492
Tuition fee	Neutral	5,643
Support staff	Positive	5,554
Teaching	Negative	1,001

Table 4: Output of processing.

Stage 3: Testing and Validation

Testing and validation starts with unit testing, followed by validation with an SME for domain- and problem-specific feedback.

The accuracy of sentiments extracted is measured by manually verifying extracted sentiment output against expected sentiment values for each of the comments.

If S is the number of text comments in the sample, and C is the number of cases in which the correct sentiment was extracted, then the accuracy of the sentiment extraction (expressed as a percentage) is C/S x 100.

Based on our experience in dealing with multiple problem scenarios across various customers and their typical acceptance criteria, we consider accuracy of at least 80 percent to be good; accuracy at 90 percent or above is excellent.

Unit testing is based on scientific sampling techniques because testing the entire, huge volume of sentiment output is often too resource-intensive to be practical. Stratified sampling techniques offer a satisfactory representation of the entire population. Testing and validation can be iterated until we achieve satisfactory coverage and accuracy.

This often calls for a change in the way important terms are extracted. Having an SME who knows the business and its problems validate the sample output is essential at this stage.

Stage 4: Processing

In the processing stage, we direct the engine to scan the entire text corpus using the fine-tuned semantic repository to extract sentiments expressed on core phrases. Table 4 shows a portion of the sentiment analysis output for a university.

Stage 5: Post Processing

After using these techniques to derive a reasonably accurate result, the results of unstructured data analysis can be combined with other structured data analysis and reported in the form of a dashboard or other visualizations, enabling interactive analysis of the combination of structured and unstructured data. (See the visualization section of this article for output samples.)

For example, sentiment analysis on Facebook data could produce specific sentiments about specific categories, which can be reported in the context of other parameters from structured data such as geography, gender, and other user profiles.

This enables "slicing and dicing" the sentiment outputs across geography, gender, and other characteristics to gain deeper insight.

Addressing Nuances of the Text Source

In addition to the complexities of linguistic analysis of sentences, there are many more complexities in identifying useful information from unstructured text data. The text corpus has many contributors, each with his or her own style of writing.

Complexities can arise from spelling errors, punctuation errors, non-linguistic entities, grammatical errors, unconventional sentence formation, short words, SMS language, and slang, among other factors. Challenges can also arise from the use of emoticons, sentences with multiple concepts and multiple sentiments, comment threads, anaphoric references in threads, double negatives, and so on.

Many unstructured data processing tools provide a method to tackle spelling errors, short words, and basic entities such as date, time, phone number, ZIP code, e-mail address, currency, HTTP address, and SMS abbreviations. Sometimes we may have to customize this further for country-specific, geography-specific, or other specific needs. For example, in some countries, ZIP codes are only numeric and always placed after a city or town name, and a specific information rule may have to be written to extract this information.

Here are a few techniques we have used to handle the additional complexities involved in processing such text data.

Emoticons

Emoticons are common in chats and comments in social media data. We have created a library of emoticon characters and their associated sentiments, and some tools offer built-in emoticon readers. However, the analysis may not always be reliable. For example, systems usually cannot detect the implied sentiment in sarcastic remarks.

Ambiguous Sentences

Ambiguity can be a challenge. For example, we have often encountered a sentence like the following for the sentiment analysis of a university: "I don't think having special class every day is a bad idea." The sentence looks simple, but can be confusing to a text analysis engine. The initial analysis showed only the negative sentiment by considering the core phrase "bad."

We addressed this by adding a rule to the semantic repository that negative verb phrases (do not, did not, does not, and so on) along with negative core phrases (bad, worse, and so on) express positive sentiment.

For a different text analysis solution, this ambiguous sentence occurred frequently: "Had this restaurant been located in the corner, it would have been very convenient." Such sentences are not simple for any text analysis engine to handle. By default, there was positive sentiment expressed for the restaurant by considering the word "convenient." In reality, the sentence actually expresses negative sentiment.

> In addition to the complexities of linguistic analysis of sentences, there are many more complexities in identifying useful information from unstructured text data. The text corpus has many contributors, each with his or her own style of writing.

Our solution was to modify the semantic repository for the phrases "would have," "could have," and "should have" along with any positive sentiment words to point to negative sentiments.

Threads in Social Media Sites

Threads in social media are chains of comments from various global users occuring across a period of time. For example, responses to a Facebook comment can be posted after one or more weeks. These threads are normally about a single core phrase, but they can branch out into multiple concepts and sentiments. It is important to note the thread ID and analyze the entire thread as a unit.

Visualization

The visualization of unstructured analysis output is a key method to enable deeper insight. Visualization of unstructured data (along with other structured data in

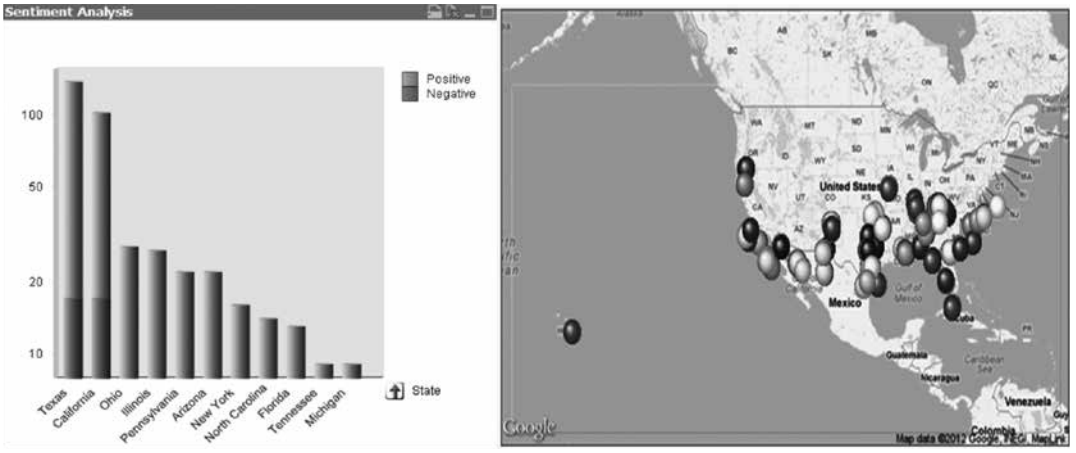


Figure 2: Visualization of first-level analysis.

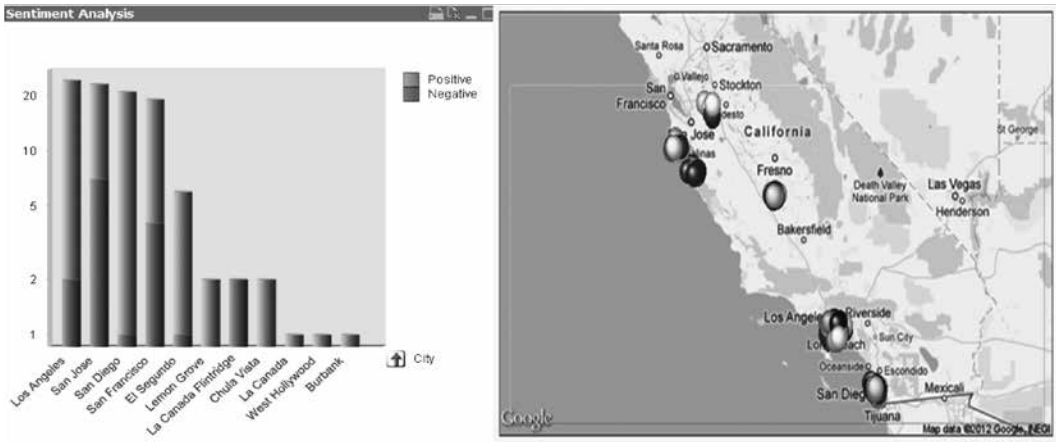


Figure 3: Visualization of the same data at a deeper level.

context) makes the graphic representation more valuable (Inmon and Nesavich, 2007).

The first-level output in Figure 2 shows a location-based sentiment analysis of a restaurant chain in North America. Users can see the total number of positive and negative comments about each restaurant. The input comes from a restaurant review website.

Figure 3 shows the second-level output; this gives the locations of the authors of the sentiments. It now becomes evident that there are more dissatisfied customers in a particular city (San Jose) than in other cities.

Summary

In this article, we have outlined an approach for significantly improving the accuracy and thoroughness of category and sentiment extraction. This entails applying a layered repository of semantics on top of the out-of-the-box sentiment analysis engine and defining and using specific concepts, entities, and categories for the problem domain. Preprocessing steps play a major role in this improvement, as does using scientific sampling techniques to understand text sources.

We have observed an increase in accuracy (moving from 50 percent to nearly 85 percent) in specific implementa-

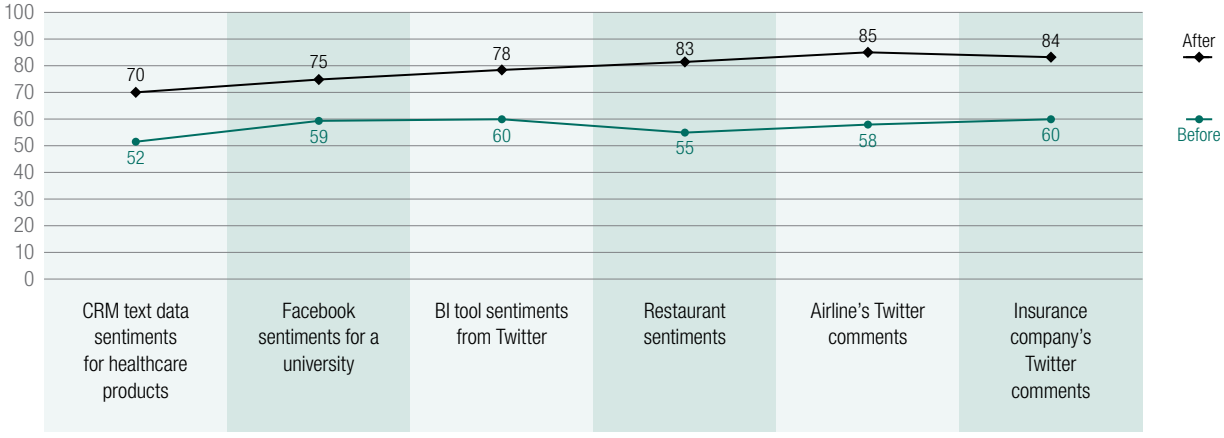


Figure 4: Improved accuracy of sentiment analysis using our methodology.

tions of this approach. (See Figure 4.) In addition, the relevance of the extracted results was high when we ensured that categories of interest were specifically defined and extracted. ■

References

Curras, Emilia [2010]. *Ontologies, Taxonomies and Thesauri in Systems Science and Systematics*, Chandos Publishing.

Feldman, Ronen, and Ido Dagan [1995]. "Knowledge Discovery in Textual Databases (KDT)," *Proceedings of the First International Conference on Knowledge Discovery and Data Mining* (KDD-95), AAAI Press.

Inmon, William H., and Anthony Nesavich [2007]. *Tapping into Unstructured Data: Integrating Unstructured Data and Textual Analytics into Business Intelligence*, Prentice Hall.

Varadarajan, Sundar, Kas Kasravi, and Ronen Feldman [1999]. "Text-Mining: Application Development Challenges," *Proceedings of SGAI International Conference on Artificial Intelligence*, Cambridge, England, Springer-Verlag.