TEXT MINING FOR CUSTOMER RELATIONSHIP MANAGEMENT: 
AN APPROACH BASED ON LATENT SEMANTIC ANALYSIS 
AND FUZZY CLUSTERING

ABSTRACT

In most CRM (Customer Relationship Management) systems, information on customers is made of structured data and it does not include significant textual information. Yet, significant textual information flows through many channels (e-mail, calls, rep notes, interviews). To analyze the textual information contained in textual documents, we propose a Text Mining methodology, that has the objective of analyzing texts without a previous knowledge of the semantic domain, based on unsupervised statistical techniques.

In our framework, Text Mining is a phase of business intelligence that distills structured data from textual documents. Specifically, Text Mining creates a model of the whole collection by means of Latent Semantic Analysis (LSA), and it identifies, through Fuzzy Clustering (FC), the relevant themes, providing a measure of the semantic content of documents. At this point results are aggregated by customer, identifying areas of his or her specific interest, and can be integrated with other information, to extend the analysis base of CRM systems.

The approach integrates LSA and FC with new elements: coordinates transformation in order to use LSA output in FC and conversion of the probabilities stemming from FC in comparable scores. The system has been implemented on a commercial product and tested with forum messages.

INTRODUCTION

Text Mining, also known as KDT (Knowledge Discovery in Texts), dates back, as a specific research field, to 1997 and it can be considered as an evolution of Information Retrieval and NLP (Natural Language Processing). While Information Retrieval aims at finding information in text to enhance the vision of the researchers, Text Mining aims at automatically extracting knowledge from textual information.

A variety of Text Mining techniques are used depending on the purposes of the research. Generally, for the classification of documents, Latent Semantic Analysis is the most used. Research studies are developing and tuning algorithms (e.g. de Freitas, Barnard 2000; Dhillon, Modha 1999).

Our work applies existing techniques to the domain of CRM, thus extending the current field of use of Text Mining.

CRM (Customer Relationship Management) systems process the events of the entire cycle of relationship with customers, from marketing to after-sale service. A specific class of CRM applications, called Analytical CRM, provide the business intelligence needed to create and analyse information on customers. Generally, customer information includes socio-graphical data, transactions and interactions with the customer, such as calls, claims, requests of information and mails. This information is stored in data warehouses and processed for a variety of business purposes, e.g. to profile customers in terms of profitability, potential, attitude, and behaviour. Thanks to these analyses, corporations have a much better...
knowledge on their customers and can better target market actions or provide more comprehensive information to customer care clerks.

Nowadays, analysis based on textual information is almost absent from Analytical CRM applications. However, textual information is relevant for a complete knowledge of the customer. First, it concerns critical relations between the corporation and the customer, such as claims, requests for information or support, answers to the questions of a salesman. Second, it depicts the issues raised by the customer on products, services or processes.

Text Mining extracts, without previous classification of text (i.e. mines), structured data from textual documents, and it classifies texts in a semantic space, so that the analysts can identify the concepts the customers refer to.

Generally, the scope of our work is unformatted textual information. Our purpose is to process this information in order to classify terms and concepts and, by this additional information, to extend the knowledge on customers.

Therefore, we have first defined a reference framework, in which Text Mining is positioned as a module of a more comprehensive Analytical CRM system. Second, we have defined Text Mining methodology that includes different steps, each one performed by specific techniques. Specifically, Latent Semantic Analysis (LSA) analyses text and Fuzzy Clustering (FC) identifies relevant concepts (clusters of issues raised by customers). These statistic-symbolic techniques do not consider grammatical or linguistic aspects but deal words as symbols.

Our choice reflects some assumptions. First, statistic-symbolic techniques are consistent with the typical characteristics of CRM texts, mainly extracted from e-mails or phone calls, that are numerous, short, often grammatically wrong, and not rigorous. Second, statistic-symbolic techniques require only a basic knowledge of the language and of the topics; therefore, you can implement and operate a mining process without a previous knowledge of the semantic and conceptual domain of texts.

ARCHITECTURE

In our view, Text Mining is an additional module of a generic Analytical CRM, as shown in Fig.1. This generic framework is used by a number of commercial products, to which a Text Mining module can be added.

Figure 1. Text Mining in Analytical CRM systems.
The core of the system is a Data Warehouse: data come to Data Warehouse from two different processes that concern, respectively, structured information and unstructured texts.

Structured information is collected from operational systems and processed to obtain a uniform input to the Data Warehouse, through a set of applications called ETL (Extraction Transformation Loading). Unstructured texts are collected by a variety of systems and, after pre-processing, go through a specific step performed by Text Mining. Text Mining extracts numerical and structured data from texts; data on individual documents are eventually aggregated by customer.

Data Warehouse receives information from ETL and Text Mining, that is subsequently processed by business intelligence applications, such as Data Mining, OLAP (On Line Analytical Processing) and reporting systems.

Within the generic framework of the Analytical CRM, we have identified a number of generic functional components of the Text Mining module (Fig.2), that operate asynchronously:

- **Input**: from document collection the system receives documents and information required to define appropriate context, and identifies the customer, if applicable, to which the document belongs.
- **Pre-processing**: standardizes documents to get homogeneous input for the Document Warehouse.
- **Document Warehouse**: stores documents, related metadata and the other data, processed by Text Mining software, that are related to individual documents.
- **Text Mining**: processes documents and returns extracted information to Document Warehouse.
- **Projection**: documents in Document Warehouse are grouped according to the customer they relate to in order to load the Data Warehouse.

In the following paragraphs we describe an experimental implementation of the Text Mining component.

**TEXT MINING LOGIC**

The Text Mining process has been implemented on an experimental version of SAS/Enterprise Miner™ 4.1. Some steps have been implemented on SAS Code nodes with SAS IML (Interactive Matrix Language). The system develops the classification model through a sequence of nodes we illustrate in the following pages (Fig.3).
The node **Import** takes the texts from an **Import** database, which stores documents split into two groups, called, respectively, **Model Documents** and **Additional Documents**. **Model Document** includes documents that are used to build the initial model through the process we illustrate in this paragraph. **Additional Documents** are an additional input that is classified within the same model of **Model Documents**, by a specific process that is described in a later paragraph. This twofold classification reflects the practical concern of avoiding a complete recalculation each time an individual document is added. Of course, a complete recalculation can start every time it is needed.

Texts are analysed by a statistic technique known as Latent Semantic Analysis (**LSA**) (*Deerwester, 1990; Landauer et al. 1998*) that considers text as a set of terms (bag of words) and obtains a $k$-dimensional space (**semantic space**) in which documents are viewed as points and dimensions represent concepts contained in the collection (**Fig.4**).

**LSA** is based on assumption that a latent semantic structure exists in a document and it can be extracted; the relations on expected word use in discourse passages are automatically inferred. **LSA** correlates terms by their presence in the documents; hence, it works better with shorter texts and larger samples.
(over 1,000 documents): this is precisely the case of textual customer communications, which actually often are in large samples and are short. In this context, LSA, that does not require a previous knowledge of the semantic content of text, is preferable to a linguistic analysis that instead asks for it. LSA includes Text Parsing and SVD.

**Text Parsing**

This step considers the occurrences of terms in each document. The calculation neglects *Stop Words*, that are words with no semantic relevance (e.g. articles). Note that a *Stop Word List* in Italian counts over 500 terms. The output is a matrix $X(t \times d)$, where $t$ is the number of different words in the collection, while $d$ is the number of documents.

**SVD (Singular Value Decomposition)**

This step, that has as input the frequency matrix $X$ computed by Text Parsing, performs matrix decomposition.

First, individual cells are weighted by a value that identifies semantic value of the term within the document, i.e. the term attitude to be used to make a sound classification. This value is the *inverse document factor* (IDF), defined for term $i$ as:

$$\text{IDF}_i = \log \frac{N}{n}$$

where $N$ is the number of documents in the collection and $n$ is the number of documents in which term $i$ appears.

Second, IDF is multiplied by the frequency of the term in the document, assuming that term utility for classification is inversely proportional to its frequency. $X$ decomposition gives 3 matrixes:

- $T(t \times k)$: associates each term to each concept;
- $S(k \times k)$: scaling matrix;
- $D(k \times d)$: associates each document to each concept.

$k$ is the number of concepts to keep, and in some research studies it ranges between 100 and 300 (e.g. Gotoh, Renals, 1997; Dumais, 1990). As result, $k$ coordinates are assigned at each document, identifying its position in the semantic space.

In order to compare documents you should define a metrics in the created space: some research studies (Landauer, Dumais, 1997; Foltz et al. 1998) consider the cosine measurement among vectors as the best metrics. To identify similarity among documents, indeed, the rate of presence of individual concepts turns out to be a better measure than the absolute value.

**Space Transformation**

In order to simplified the concept, the modulus of the vectors that represent the documents are normalized by using Euclidean distance: in this way the sequence among vectors does not change, in respect to the sequence which is obtained using a metrics based on the angle value. The use of the Euclidean metrics to normalize coordinates does not affect the order obtained by taking a document and by sorting the other ones on distance, and it changes the distance values only marginally. If you consider linear transformation and compare the Euclidean distance to $d^2=1-\cos(\theta)$, you can see in Fig.5 how small is the gap between the distances computed between two documents placed anywhere in the space (the angle is always measured in the bidimensional plane which contains the two vectors).
**Figure 5.** Comparison between the distance of two documents computed using the Euclidean distance and \( d_2 = 1 - \cos(\theta) \).

**Fuzzy Clustering**

To the resulting space a Fuzzy Clustering (FC) algorithm is applied, within **Fuzzy Clustering** node: clusters centers are identified in the space *(Fig.6)*, while at each document is assigned a membership probability for each cluster.

FC can avoid some problems arising from Boolean clusters, when multiple themes are present in the same document. Actually, a document with multiple themes cannot be rightly classified within Boolean clusters; by contrast, within fuzzy clusters, it would be placed in the middle of the centers of the clusters and it would be associated with scores that measure the presence of the relevant themes. Furthermore, you can assign a continuous value to the membership of documents to clusters, based on the distance from the respective center, and you get a more precise classification.

Resulting clusters are organized not hierarchically, because the system neglects the relationship among themes while it considers the relationship between themes and documents (and between themes and customers).

**Figure 6.** Center of a cluster in the semantic space.
We propose to use an Extended fuzzy c-means clustering \cite{Kaymak2000} algorithm, in which cluster centers are extended in the space to form hyper-spheres and in which the number of clusters is automatically determined through an aggregation process. Algorithm parameters are:

- initial number of cluster;
- termination criterion;
- fuzziness parameter.

Fuzziness parameter is especially important to assure valid results, while the other parameters can have standard values. As result of this step, at each document is assigned a membership probability to each cluster.

**Scoring**

With the results of the previous node, we can compare the presence of each theme within a document, but we cannot compare documents across the sample. Therefore, we need to go from a relative probability metrics (where the sum is always equal to one) to a metrics that weights the degree of importance of each subject within the whole sample of documents. The choice of this metrics is critical; a simplistic use of probabilities may cause some logic paradoxes, which prevent from comparing documents. Let us consider:

- a document A that deals only with a given theme (x);
- a longer document B, that contains the document A and that deals with an other theme (y).

Let us suppose that, in relation to cluster x, A has a score of 1, while B has a score of 0.5 (we further assume B treats both themes x and y at the same degree). The probabilities cannot be reciprocally compared, because it would result document A deepens theme x more than document B. To overcome the paradox, we have to make a probability transformation.

Probabilities are weighted by an index of the semantic relevance of the document, i.e. the degree of deepening of the subjects it deals with. Based on experimental considerations, we propose to use a semantic importance index associated to each document defined as the modulus of the vector representing the document in the semantic space before normalization. The higher number of included terms, the greater their semantic meaning, the higher the correlations with concepts and, therefore, the larger the modulus of the vector will be.

In the case above, experimental evidence shows the modulus of vector B is roughly twice the modulus m of vector A: by weighting computed probabilities with modulus, we obtain for the cluster x the next values:

- A: \( 1 \times m = m \);
- B: \( 0.5 \times 2 \times m = m \).

In this individual case the two values are, as wished, equal. Though a perfect correspondence is rare, we can assume an acceptable approximation in getting comparable scores that identify the degree of deepening of a given theme in different documents.

**Description**

In order to support an analysis, we should give a semantic description to each cluster through a list of relevant words. Description node calculates for each cluster and for each word a significance index, that defines the aptitude of the term to describe the cluster (considered as a point in the semantic space),
considering correlations between clusters centers and concepts and then between concepts and terms. Normalization of coordinates does not affect these results because only the comparison among different indexes, not their absolute value, is important.

Starting from terms with greater index, only the terms which indexes summation does not exceed a predefined parameter, called **total significance index**, are selected. As result, to each cluster is associated a word list, used by analyst to understand the meaning of the semantic space.

**PROCESSING OF ADDITIONAL DOCUMENTS**

Model calculation is expensive. If the collection of documents is numerous, new individual documents can be processed by re-using the existing model, providing that they are not too semantically far from existing sample. The re-use of model avoids an expensive complete recalculation of the whole collection of documents. To describe this shortened recalculation path, we have conceived an insertion route for additional documents that assumes their logic is consistent with the existing model structure. The configuration of the nodes, as presented by SAS Enterprise Miner™, is in Fig. 7.

![Figure 7. Text Mining general system in SAS Enterprise Miner™.](image_url)

- **Text Parsing** node computes the frequency of the words included in the new documents, using the same *Stop Word List* as the standard **Text Parsing** node.
- **Space insertion** node considers only the terms present in the original collection to obtain a frequency matrix which, multiplied by the matrix used by **SVD** node, gives the position of each additional document in the semantic space.
- Coordinates are now normalized and, through **Clusters insertion** node, membership probabilities for each cluster are calculated.
- Results are multiplied, in **Scoring2** node, by the **semantic importance index** and the scores, together with the scores obtained from the other **Scoring** node, are joined and exported in the **Results** database.
OUTPUT

System output includes scores and description of clusters.

Scores measure how much a document is close to a cluster of given topic. The scores of the documents can be analysed by a business intelligence application, in order to identify relevant topics and their trend over time. Scores related with identified documents are aggregated by customer, adding scores related with each subject: scores are additional attributes (facts) of a customer within the information stored in the Data Warehouse.

Of course, raw scores need to be transformed in indexes that are meaningful to a customer analysis strategy. For example, refined scores can be used together with standard numeric data for profiling individual customers or for defining a promotion campaign. In the first case, the output is a set of scores that describe the interest of an individual customer on a set of issues. In the latter case, the outputs are clusters of customers.

Cluster descriptions are lists of words that semantically describe each cluster. These data can be used for general analysis (focused on clients) and serve as further base for the construction of predictive and descriptive models by Data Mining system.

AN EXAMPLE

The system has been tested with a collection of messages from an Internet discussion forum about the products of a mobile phones vendor. These documents are very similar to e-mails that are a very common message in CRM.

The test sample accounts 1,100 documents in Italian, that include 8,407 different terms (excluding Stop Words), with an average length of 35 terms in each message. 1,000 documents were used for model construction, while the other 100 documents have been inserted in the built model. The values of used parameters are:

- dimensions of semantic space: 150;
- initial number of cluster: 30;
- termination criterion: 0,001;
- fuzziness parameter: 1,2;
- total significance index: 1.

Text Mining system has determined 13 clusters.

System performance is comparable to analogous classification systems that use LSA (Landauer et al. 1998; Dumais et al. 1988). The system has classified correctly about 70% of the documents.

To give an example of the system logic, let us take a short document, labelled as 412, taken from test sample and translated into English.

Hi! I’ve got some connection problems between my GPRS mobile and my palm through IRDA. Which driver I must use for the modem and for the palm? Thanks
After tokenization and *Stop word* elimination, this is the result:

*connection problems GPRS mobile palm IRDA driver must use modem palm*

After **SVD**, in *Tab. 1* are visible some coordinates of the document in the semantic space (multiplied by 1,000).

<table>
<thead>
<tr>
<th>D1</th>
<th>D2</th>
<th>D3</th>
<th>D4</th>
<th>D5</th>
<th>D6</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>412</td>
<td>9,622</td>
<td>-4,44</td>
<td>4,618</td>
<td>-1,19</td>
<td>-1,50</td>
<td>5,746</td>
</tr>
</tbody>
</table>

*Table 1. Coordinates of document 412 in the semantic space.*

After normalization, FC and scoring, 13 scores are associated with the document (*Tab. 2*).

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>412</td>
<td>0,026</td>
<td>0,021</td>
<td>0,025</td>
<td>0,026</td>
<td>0,023</td>
<td>0,136</td>
<td>0,025</td>
<td>0,028</td>
<td>0,024</td>
<td>0,017</td>
<td>0,035</td>
<td>0,024</td>
</tr>
</tbody>
</table>

*Table 2. Scores for document 412.*

We can see the higher value associated with cluster number 6. The description of this cluster shows well the semantic contents of the sample document (*Tab. 3*).

<table>
<thead>
<tr>
<th>CLUSTER</th>
<th>WORD</th>
<th>INDEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>modem</td>
<td>0,077</td>
</tr>
<tr>
<td>6</td>
<td>driver</td>
<td>0,048</td>
</tr>
<tr>
<td>6</td>
<td>irda</td>
<td>0,048</td>
</tr>
<tr>
<td>6</td>
<td>gprs</td>
<td>0,044</td>
</tr>
<tr>
<td>6</td>
<td>connection</td>
<td>0,037</td>
</tr>
<tr>
<td>6</td>
<td>mobile</td>
<td>0,026</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

*Table 3. Description of cluster 6.*

At this point, we can use mining results in a more general analysis; for example, we could measure the opportunity to offer some new service, and use the scores by client to identify customers potentially interested to the new service.
However, if we consider all the scores on the sample and examine the top score of each document, we see that many documents are poorly classified, because they are semantically poor or concern rare themes (Fig. 8).

![Figure 8. Values of the greater score for document in the collection.](image)

**CONCLUSION**

Text Mining can extend the knowledge on customers in Analytical CRM systems. Our approach addresses the profile of the customer, and, therefore, adds the knowledge arising from textual information to the profiling of customers, that is currently built on numerical information.

Our methodology is oriented to large collections of short, unformatted and unstructured messages. Therefore, it uses statistical techniques: Latent Semantic Analysis has the advantage of building a classification from scratch while Fuzzy Clustering avoids pitfalls of traditional clusters. These techniques have been enhanced by some additions conceived for the CRM domain. Specifically, we propose to normalize the coordinates that are identified by Latent Semantic Analysis and to weight the probabilities identified through Fuzzy Clustering, in order to define an absolute metrics to compare documents and to avoid potential inconsistencies in analysis.

In our view, our work is a step in the research of CRM Analytical Systems. Further research can include a specific assessment of benefits arising from Text-Mining-based indexes and the evaluation of usability, scalability and performance of CRM-oriented Text Mining systems.

**REFERENCES**


