UNIVERSITÀ DEGLI STUDI DI CATANIA

Department of Mathematics and Computer Science

Master Degree Course in Computer Science

Michele Ferro

Knowledge extraction from sustainability reports using computer vision-based heuristics

Final Project Report

Supervisor: Giovanni Gallo

Co-supervisor: Salvatore Nicotra

Academic Year 2022 – 2023

Abstract

The exponential growth of digital documents in recent years has presented a significant challenge and opportunity for knowledge management. This thesis explores the field of knowledge extraction from documents with specific applications in the realm of sustainability. It delves into various techniques, methodologies, and technologies developed to extract valuable information and insights from unstructured textual and visual data.

The research investigates the diverse applications of knowledge extraction, ranging from information retrieval and recommendation systems to data mining and natural language processing.

Contents

1 Natural Language Processing fundamentals

1.1 General overview

In these times, more and more documents of various genre are published, from scientific papers to financial and corporate reports. In turn, the informations that these documents provide should presumably offer a service to the research, to a third-party client or to the corporate releasing the report itself.
However, being them source of **unstructured data**, the information they contain cannot be actually exploited because of the natural language they are characterized by, which is misunderstandable and difficult (for a machine) to represent.

In order to make use of unstructured data, it is mandatory a preventive **knowledge extraction** operation, the advantage of which is to define a set of "interested indices to catch", formalize and finally quantify them in order to later enter them into a computerized system.

The most known knowledge extraction operation is the so-called ***Natural Language Processing*** (**NLP**), a set of procedures that – by using artificial intelligence heuristics – are able to understand and represent the natural language.

1.2 Use cases

Dealing primarily with text, NLP tasks mainly require the extraction of inferable context from sequences of words, such as web pages, posts or business information. Clearly, this area of data science is mainly devoted to automating simple tasks, such as language recognition, semantic analysis and sentiment analysis.
In order to perform the above tasks in automated manner, enterprises adopt various NLP tasks:

- **Text Analysis**: analysis of a text and, where required, identification of key elements (e.g., topics, people, dates);
- **Text Classification**: interpretation of a text to classify it into a predefined category (e.g., spam);
- **Sentiment Analysis**: mood detection within a text (e.g., positive/negative review);
- **Intent Monitoring**: understanding text to predict future behaviors (e.g., a customer's willingness to purchase);
- **Smart Search**: searching within archives for documents that best match a query posed in natural language;
- **Text Generation**: automatic generation of a text;
- **Automatic Summarization**: production of a summarized version of one or more text documents;
- **Language Translation**: translation of text by choosing, on a case-by-case basis, the best meaning depending on the context.

More and more enterprises are today interested in NLP solution; there are several opportunities of natural language processing systems for business:

- analysis of corporate emails (e.g., to recognize unwanted messages and sort incoming mail by topic);
- extraction of information from governance documents, such as reports and procedures, to ensure quick reference;
- projects for analyzing administrative documents, and solutions for analyzing internal company communications such as help-desk emails;

- analysis of social network posts;
- algorithms for understanding website navigation queries and redirecting search correctly;
- solutions for analyzing journalistic news, for instance to recognize fake news.

This thesis will analyze, more in particular, the application of the NLP, by using computer vision heuristics, on sustainability reports released by corporations at the end of a process that will be described in the next section.

1.3 Materiality analysis

1.3.1 What is it

The **materiality analysis** is a process that makes able to detect everything has an impact on the company's business or on which the business could have an actual impact. It is to note how the term "materiality" highlights the importance of all the elements that make clear the company's commitment on being sustainable.

To make this analysis effective, the former necessity is the stakeholder engagement, that is the involvement of all the stakeholders who influence and/or are influenced by the organization activities. Some of these are:

- employees;
- providers;
- customers;
- medias;
- population;
- environment;
- and other similar entities...

The analysis shows off the management **accountability** in different kind of capitals (finance, production, nature, social and human interactions), granting a deeper insight about their inter-dependency.

1.3.2 How it is executed

The process is made up by different steps, each one analyzing a specific behavior or strategy of the organisation to be analysed.
First of all, is necessary a closed-question questionnaire about the major topics:

- **the materiality analysis process**, gathering information about choices that corporations have made regarding the development of the materiality analysis itself;
- **managers' thoughts and judgments** with respect to changes triggered by the materiality analysis, collecting assessments of the actual impact of the materiality principle on the reporting processes undertaken by companies;
- **opinions regarding the relevance of the faced problems**, requiring the managers to express their own judgement about those problems of the materiality analysis that emerged from the conducted interviews.

Next, depending the organisation, other **sources** have to be integrated to deepen the analysis, i.e.:

- customer satisfaction survey history;
- analysis of the engagement activities with local communities, gathered along the years;
- indications made by ethical rating companies;
- meeting held along time with trade union representatives;
- social medias and press reviews examination;
- examination of the judgements expressed by the opinion leader towards banks.

Gathered all these elements, the heads of the organisation and the stakeholders can finally draw up the materiality matrix.

1.3.3 Materiality matrix

The **materiality matrix** is the final output of the entire process; typically it is a bi-dimensional plot having on the x-axis the values relevant for the **company**, and on the y-axis those which are relevant to the **stakeholders**.

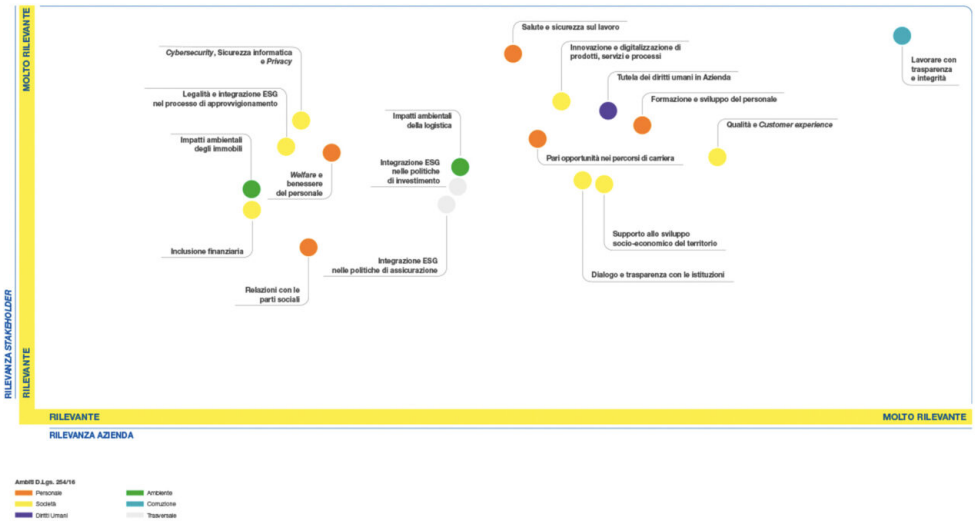


Figure 1.1: Example of materiality matrix

Substantially, by the positioning of the themes, the viewer could have a clear insight about the company sustainability objectives and commitment, and how relevant they are.

As a matter of fact, after viewing the materiality matrix, the next step is a further comparison in which all the functions involved in the analysis process and the top management try to understand the polarization values and the classification reasons: assuming this, without this particular passage, redacting a materiality matrix becomes a mere reporting exercise.

1.3.4 The lack of standardisation

Although the importance and benefit of this business process could easily be noted, the lack of a standard makes it barely actually useful; the presence of countless forms and types of these plots – joined with the lack of actual quantitative data – goes after the direction of the entire process, making difficult to understand the path the organisation has taken during the year. In order to ease the step following the analysis, this thesis' project proposes a new scalable solution thought for all the organisation.

Using the proposed system, the top heads of an organisation could be able to easily extract the plots contained in a sustainability report document, transpose them into "standardised plots", and obtain for each of them a comma separated value file that could be imported in an *Elastic-*

Search/*OpenSearch* instance. Then, using a search engine powered by this instance, the user could query the system about the gathered sustainability informations.

2 Computer vision methods involved

In this chapter will be analyzed the state-of-the-art computer vision feature-extraction algorithms that made this project possible. More specifically, they have been used to develop the extraction and interpretation heuristics that will described during the description of the proposed solution, in the next chapter.

2.1 Hough transform

Proposed by Paul Hough in 1962, the Hough transform is a method for detecting lines in images, being them geometrical objects which could analytically define human-derived shapes.
Before its birth, the state-of-the-art of the line detection consisted in two methods: the first being template matching and second one consisting in detecting every line featured by their points, taken two by two, in order to then find the subset of vertices crossed by those lines. However, the former method requires a well-known mask M, like the following one:

$$M = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$$

(**Mask defining a line's segment with zero orientation**)

while the latter requires a massive number of comparisons. In fact, being the lines featured by n vertices we have:

$$\frac{n(n-1)}{2} \approx n^2 \qquad \textbf{(Number of the lines crossing the } n \textbf{ vertices)}$$

$$n\frac{n(n-1)}{2} \approx n^3 \qquad \textbf{(Number of comparisons)}$$

By contrast, the innovative method proposed by Hough consisted in mapping a tough problem into a more feasible one: detecting peaks in the parameter space of the researched curve or, more specifically, the researched line.
Therefore, this solution could be applied in both the realms of the straight lines and the curves, granting a "general" algorithm for every kind of image.

2.1.1 Hough transform for straight lines detection

Every line defined by the equation

$$y = mx + n \tag{2.1}$$

is featured by a couple of parameters (m, n) which in turn, in the parameter space, defines a plane. Meaning that, in the parameter space the line is defined by a vertex.

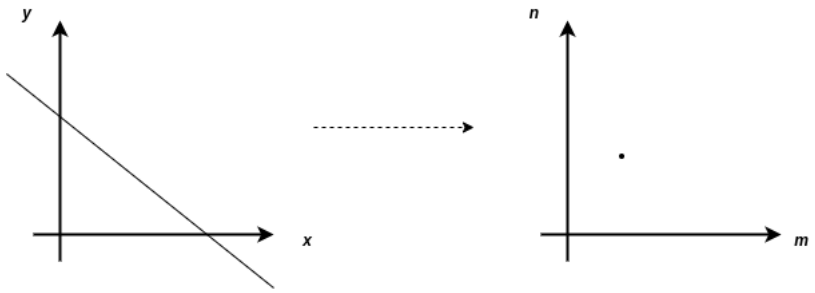


Figure 2.1: Line representation in the parameters space

On the other hand, every vertex (x, y) in the original space, represents a line n in the parameter space:

$$n = x(-m) + x \tag{2.2}$$

Every vertex of this line is to be identified with another line, crossing the vertex (x, y), in the original space.

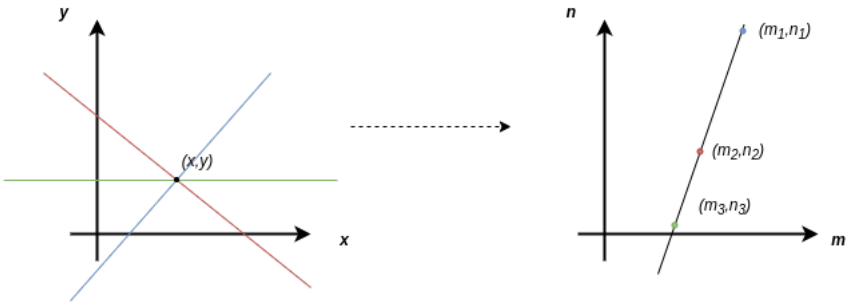


Figure 2.2: Vertex representation in the parameter space

As a consequence, two vertices belonging to the same line r, match – in the parameter space – with two lines whose intersection gives the r line's couple of parameters (m, n). Hence, a line in the original space, defined by a set of N vertices $P_1, \ldots, P_N$, is to be identified in the parameter space with the intersection of N lines, each one corresponding to a vertex P_i.

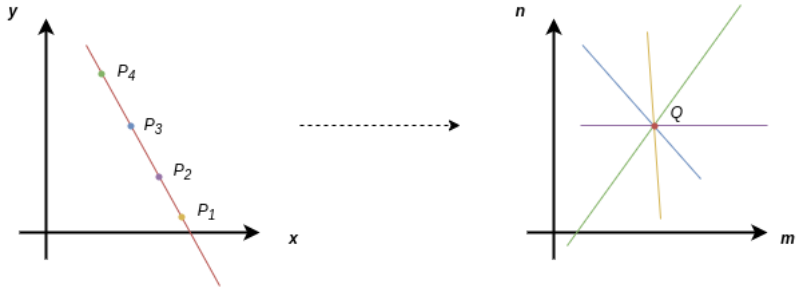


Figure 2.3: Definition of set of points, crossed by a line, in the parameter space

Nonetheless, it should be noted that the collinearity between the vertices P_i could not be assured in presence of noise; this assumed, the intersection could not be unique; having enough vertices in the original space, the problem could be transposed in the detection of peaks in the parameter space.

Following is described the algorithm of the line's Hough transform. For simplicity, it is assumed that the input image contains just one line, featured by a couple of parameters (m', n') and crossing the edge points $P_1, \ldots, P_N$.

The algorithm is composed by the upcoming steps.

- Firstly, the parameter space (m, n) is divided into a grid composed by a discrete number of cells, to each of which it is associated a counter $C(m, n)$.
  Then, for each vertex $P_i \equiv (x_i, y_i)$:

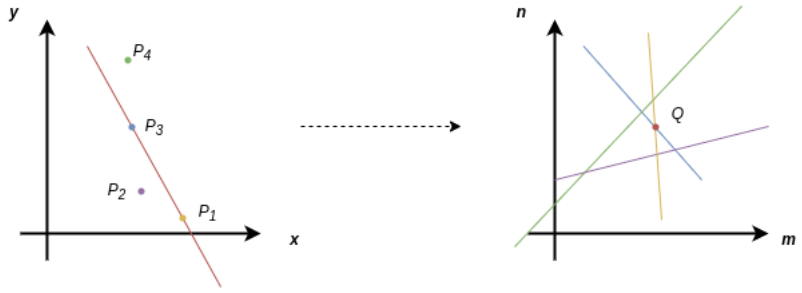


Figure 2.4: Detection of peaks in the parameter space

  - it is calculated the line s_i having, in the parameter space, the coefficients (x_i, y_i);
  - the counters relating the line s_i (in the parameter space) are incremented.

- In absence of noise, every s_i crosses the cell (m', n'); so, $C(m', n') = N$ is the peak.
- Finally, through this "voting process", the peak is identified and the line is detected.

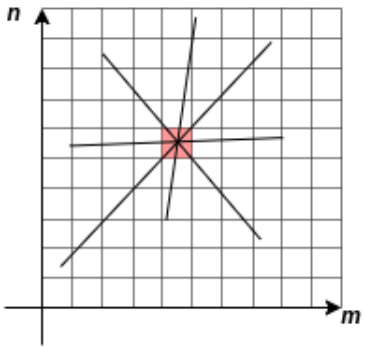


Figure 2.5: Accumulator defined on a 2D-grid: here, $C(m, n) = N$ where the cell is red (maximum number of intersections)

Although this method is fast and computationally easy, it is defined on a discrete number of parameters; being this space infinite, the procedure's implementation requires a feasible way to set a maximum and a minimum value to n and m.
This could easily be done by choosing an alternative definition of the lines,

like the following one.

$$x \cos\theta + y \sin\theta = r \tag{2.3}$$

Here, r defines the distance between the line r from the origin and its heading $\theta \in [0, \pi]$.

Figure 2.6: Representation of line by using the heading

Given that the vertices in the original space belong to an image, r belongs to a discrete range; in fact, $r \in [0, \sqrt{M^2 + N^2}]$.

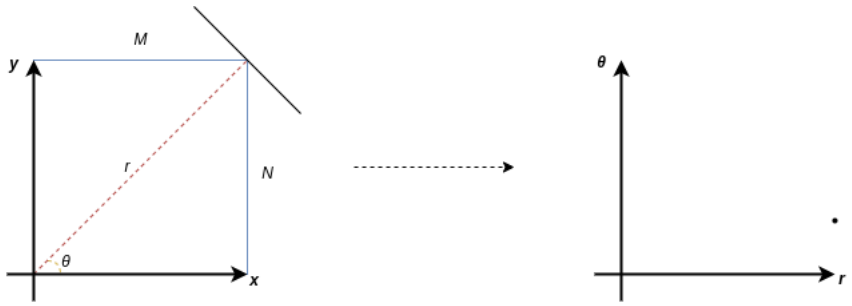


Figure 2.7: Representation of a $M \times N$ image by using the heading

In conclusion, each point matches, in the parameter space, with a curve to the family of lines passing through that particular point.

2.1.2 Circular Hough transform

Every edge could by generalized by the equation

$$f(x, y, a_1, a_2, \ldots, a_n) = f(x, y, \tilde{a}) = 0 \tag{2.4}$$

where $\tilde{a}$ is the curve's parameters vector. Therefore, the procedure described in the previous section could be adjusted in order to deal with the research of curve edges.

More specifically, the new procedure could be summarized by the following steps.

- First of all, the parameter space $(a_1, a_2, \ldots, a_n)$ has to be opportunely quantized by defining a cumulative matrix $A(a_1, a_2, \ldots, a_n)$, the coefficients of which are initially set to zero.
- Following, for every image's pixel (x, y) having maximum value (or having higher value than a previously set threshold), the value of every cumulative matrix's cell satisfying the parametric equation, according to the parameters a_i defined during the previous step, is incremented by a unit.
- Finally, the cumulative matrix A is analyzed: every meaningful peak $A(a'_1, a'_2, \ldots, a'_n)$ is a candidate for the representation in the domain space of the original curve $f(x, y, a'_1, \ldots, a'_n)$.

This procedure could also be used for detecting circular shapes.
Let a curve defined by the equation:

$$(x - a)^2 + (y - b)^2 = r^2 \tag{2.5}$$

where the parameters space is tri-dimensional as the tuple is composed by the three parameters (a, b, r). If the radius r is known, actually its research can be reduced to 2D, aiming to find the (a, b) coordinates of the circle's center. So:

$$\begin{aligned} x &= a + r\cos\theta \\ y &= b + r\sin\theta \end{aligned} \tag{2.6}$$

The locus of the points having (a, b) coordinates in the parameter spaces fall on a circle of radius r centered at (x, y). The true center point will be common to all parameter circles, and can be found by using the aforementioned "voting process".

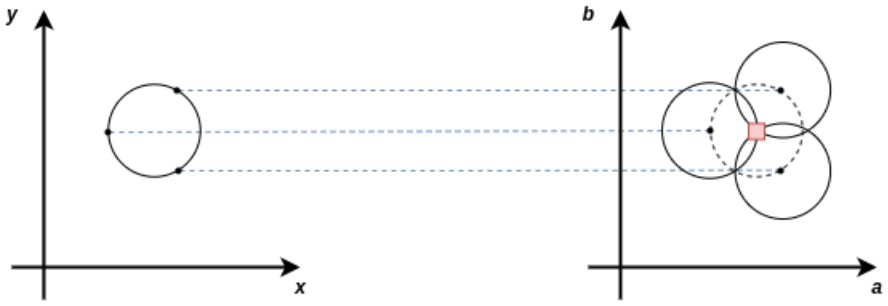


Figure 2.8: Representation of a circle in the parameter space

2.2 Blob detection

Through blob detection methods it is possible to detect circular regions in a digital image that differ in properties (color, texture, brightness and so on) compared to surrounding regions; this assumed, a blob is a circular region of an image in which some properties are (approximately) constant.
The idea behind blob detection is to detect circular regions by convolving the image with a multi-scale blob filter, in order to then search for filter response extremes in the resulting scale-space.

2.2.1 The blob filter

The base of the blob filter is the ***Laplacian of Gaussian*** (***LoG***), a circular and symmetric operator made up by two additional operators.

- The ***Laplacian***: a 2D isotropic measure of the 2nd spatial derivative of an image. It highlights regions of rapid intensity change and is often user for edge detection (the most widely known application is, in this regard, the *zero-crossing edge detector*). Given an image with pixel intensity values $I(x, y)$, it is given by:

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{2.7}$$

  It can also be discretely approximated by a mask similar to the following one:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{2.8}$$

- The ***Gaussian smoothing filter***: a 2D convolution operator used to "blur" an image in order the remove noise.

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(- \frac{x^2 + y^2}{2\sigma^2} \right) \tag{2.9}$$

Because the *Laplacian* kernel approximates a 2nd derivative on the image, as previously stated, it is very sensitive to noise: to counter this, the *Gaussian* smoothing filter is applied on the image before convolving it with the *Laplacian*.
Therefore, as the convolution operation is associative, the *Gaussian* filter can be firstly convoluted with the *Laplacian*, and then the resultant filter can be applied to the image.

The 2D *LoG* function centered on zero and with *Gaussian* standard deviation σ has the form:

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right]\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{2.10}$$

also represented under the form:

$$\nabla^2 G = \frac{\partial^2 G}{\partial x^2} + \frac{\partial^2 G}{\partial y^2} \tag{2.11}$$

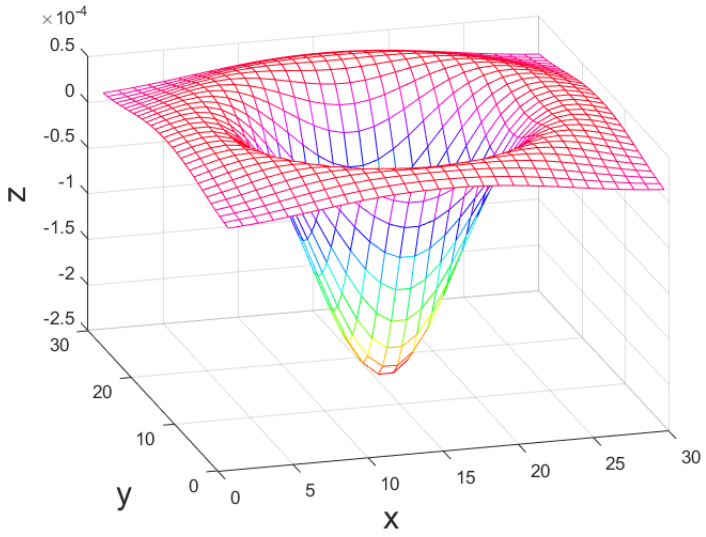


Figure 2.9: Graphical representation of the *LoG*

The application of this convolution results in strong positive responses for dark blobs of radius $r^2 = 2\sigma$ and strong negative responses for bright blobs of similar size. However, the major problem given by this operator is that its response strongly depends on the relationship between the size of the blob structures in the image domain and the size of the *Gaussian* kernel used for pre-smoothing. Hence, in order to detect blobs of different unknown size in the image domain, a multi-scale approach is necessary.

A straightforward method consists in normalizing the scale by multiplying the derivative by σ; as the *Laplacian* is the 2nd derivative of the *Gaussian*, it is therefore multiplied by σ^2. The normalized-scale filter is then represented by:

$$\nabla^2_{\text{norm}} G = \sigma^2\left(\frac{\partial^2 G}{\partial x^2} + \frac{\partial^2 G}{\partial y^2}\right) \tag{2.12}$$

The scale producing the peak of *Laplacian* response in the center of the blob itself is told **characteristic scale**. Knowing that, in order to get optima responses, the zeros of the *Laplacian* have to be aligned with the circle, the maximum response occurs at $\sigma = \frac{r}{\sqrt{2}}$.
Meaning that, the last step consists in finding the optima of the LoG in space and scale.

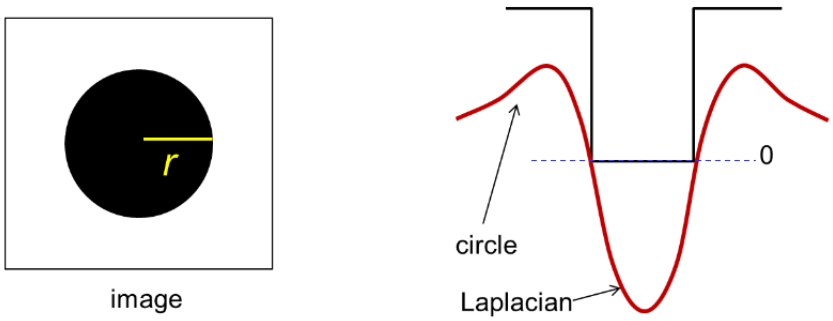


Figure 2.10: Response of the filter

2.2.2 Description of the algorithm

Summarizing the above, the blob detection algorithm is divided into the following steps, given an input image.

- At first, the image is convoluted with a scale-normalized LoG at several scales;
- then, the maxima of squared LoG response in scale-space are found.

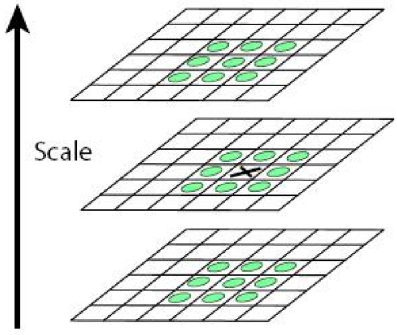


Figure 2.11: Search of the optima in the scale-space

2.3 Optical character recognition

Optical character recognition (OCR) is referred as the conversion of images of typed, handwritten or printed text into machine-encoded text. Apart from using it for data entry purposes, it is often used as an help to blind people, by reading scanned documents.

2.3.1 General OCR process

Every kind of OCR technologies acquire textual data from a document by different steps, after taking a sort of "photograph" of the document's page.

- **Image pre-processing** – The OCR software improves the elements of the document that need to be captured, by removing particles, isolated pixels and noise in order to get a plain and clear text.
- **Binarization** – The refined page is converted into a bi-level image, containing only black and white colors: therefore, black areas are identified as characters while white areas are identified as background. This contributes to apply a segmentation process to the document, to easily differentiate the foreground text from the background, and so helping the following character recognition.
- **Character recognition** – Focusing on one character at a time, the black areas are processed to identify alphanumeric characters. The recognition is carried out by using two different types of algorithms:
  - **pattern recognition**: they insert text in different fonts and formats into the OCR software; the software itself is then used for comparing and recognizing the characters in the scanned document;
  - **feature detection**: they apply rules considering the features of a certain letter or number to identify characters in the scanned document; such text recognition techniques are the basis of most deep learning OCR methods, typically based on ***L**ong-**S**hort-**T**erm-**M**emory* (***LSTM***) networks.
- **Post-processing** – The output of the OCR system is finally analyzed for any context-based or grammatical error (also to improve the accuracy of the system). Different strategies can be adopted to correct the output.

- **Proof reading**: even being the most obvious way of correcting the output, it can be tedious as it is done by human effort.
- **Lexicon error correction**: approach used to correct spellings in the output text by comparing a word against a dictionary of similar words.
- **Grammar/semantic error correction**: typically this is performed by calculating the *Levenshtein distance*, which is a string metric for measuring the difference between two sequences. Informally, the *Levenshtein distance* between two words is intended as the minimum number of single-character edits (insertions, deletions or substitutions) required to change one word into the other. The *Levenshtein distance* between two strings a and b (of length $|a|$ and $|b|$ respectively) is given by

$$\operatorname{lev}(a,b) = \begin{cases} |a| & \text{if } |b| = 0 \\ |b| & \text{if } |a| = 0 \\ \operatorname{lev}\big(\operatorname{tail}(a), \operatorname{tail}(b)\big) & \text{if } a[0] = b[0] \\ 1 + \min \begin{cases} \operatorname{lev}\big(\operatorname{tail}(a), b\big) \\ \operatorname{lev}\big(a, \operatorname{tail}(b)\big) \\ \operatorname{lev}\big(\operatorname{tail}(a), \operatorname{tail}(b)\big) \end{cases} & \text{otherwise} \end{cases} \tag{2.13}$$

  where:
  - the tail of some string x is a string of all but the first character of x;
  - $x[n]$ is the n-th character of the string x, counting from 0.

In the next section will be deeper analyzed the implementation that made possible this project, belonging from the class of feature detection systems.

2.3.2 Google's Tesseract OCR Engine

Tesseract, an open-source OCR engine, was initially developed at *Hawlett-Packard* (*HP*) *Labs* between 1984 and 1994 as PhD research project; its aim was to be a possible software add-on for *HP*'s line of flatbed scanners. At the end of 1994, its development ceased entirely; however, the next year the engine was sent to *UNLV* for the *1995 Annual Test of OCR Accuracy*, where it proved its worth against the state-of-the-art of that time; then, in 2005, *HP* decided to make this software open source.

Tesseract OCR implementation assumes that its input is a binary image with optional polygonal text region defined, as *HP* had an independently-developed page layout analysis technology that was used in the products of the time (obviously, this was a closed-source technology).
Even following a traditional step-by-step pipeline, *Tesseract OCR* featured unusual methods (even now) that made it stand out from other OCR engines. The processing pipeline is summarized as it follows.

- Firstly, a connected component analysis is performed; here, the components' outlines are stored. The authors state that, even being a computationally expensive design – at least, at that time – it had the advantage to simply detect white-on-black text and recognise it as easily black-on-white text, making *Tesseract* (probably) the first engine able of handle such a thing. At this stage, outlines are gathered together, purely by nesting, into *Blobs*.

- The output *Blobs* are then organized into text lines, which are then broken into words differently according to the kind of a character spacing.

- Then, the recognition step proceeds as a two-pass process:

  – firstly, is attempted to recognize each word in turn; in particular, those words being satisfactory are passed to an adaptive classifier as training data, which than gets a chance to more accurately recognize text lower down the page;

  – then, a second pass is performed as the adaptive classifier may have learned useful information too much later to make a contribution in the "higher" section of the page.

- Finally, fuzzy spaces are resolved and small-cap text is located by checking alternative hypotheses for the x-height.

3 Proposed solution and implementations

The implementation of the following solutions is available via the repository `https://github.com/nebuchadneZZar01/sustineo_extractor`.

3.1 Visual data extraction

Having a `PDF` format file document, the first step consists in extracting the visual data containing interesting information. However, as these documents are written using the software *InDesign* from the *Adobe* suite – which uses a closed standard –, the visual data actually doesn't consist in real raster images imported onto the input document, but as a complex shape made up by simpler ones (e.g. vector lines, circles and squares) in a tree structure. For this reason, the extraction is based on the application of different computer vision algorithms, depending from the situation; moreover, this allows a more versatile approach, as it can potentially be used on every kind of input document.

It should be noted how, instead of ML approaches – based on CNNs –, have been chosen heuristic based on classic Computer Vision algorithms. This makes able to have a lightweight and fast system, without the necessity of a preventive training operation, which could result arduous because of the absence of labelled data.

3.1.1 Document pages formalization

First of all, is mandatory a formalization of the pages on which the proposed heuristics will be applied. This will be useful to understand in which area of each page, the computer vision algorithms will have an action.
In particular, it will be taken as instance an A4-format page belonging to one of the sustainability report documents used during the experimentation of this approach. However, it is to be noted that even other formats (such as A3 or A5) may be possible.

Figure 3.1: Example of page belonging to a report

It is obvious that the majority of the shapes will be concentrated in a specific region of the page; so, even if the heuristics will be actually applied on the entire page, only the elements detected on the mentioned interesting region will be later considered as input of the following analysis; this to avoid that minor graphic elements of the page, located on its furthest edges, could be detected as interesting graphics. Hence, a generic page belonging from a document could be formalized as composed by two areas:

- an **unfeasible region** (highlighted in red in **Figure 3.2**), delimited by margins of which the measures have been pre-defined both horizontally and vertically; this section contains neglectable shapes;
- a **feasible region** (highlighted in green in **Figure 3.2**), localized at the center of the page; this region contains shapes that will be subsequently analyzed.

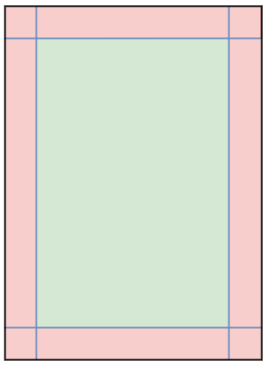


Figure 3.2: Example of page sections

3.1.2 Visual data extraction process

The behavior of the proposed approach is summarized by three subsequent passages:

1. **pages conversion**: starting from each page of the document, a couple of two images is generated in order to be used during the following steps;

2. **interested string search**: using an OCR routine, the pages containing a materiality matrix are distinguished from those not containing them, as it is specified in the page header;

3. **visual data detection and export**: depending from the type of interested visual data (materiality matrix or generic infographic/table), an heuristic based on computer vision methods is applied to detect the region containing it, which is later cropped and exported as a `PNG`-format image.

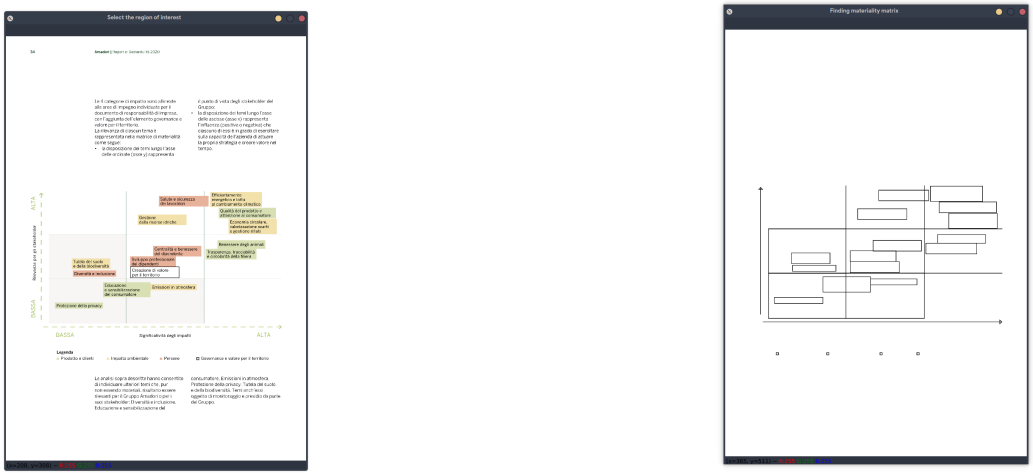
However, other possible interventions can be made on the final result.
Each one of these passages will be formally analyzed and described in the following sections.

3.1.2.1 Pages conversion

A `PDF` is a sequence of pages in which several types of vector data can be found, shapes and text among them; given this, a computer vision algorithm cannot be properly applied, as being thought to work on raster data.
The proposed approach consists in generating two different raster images for each page of a document:

- a **plain-text** page-image (**Figure 3.3a**): it is a sort of plain high resolution "photograph" of a document page, which will be later used to crop the interested region at the end of the detection;
- a **shape-only** page-image (**Figure 3.3b**): another high resolution image, but differently from the previous one it contains only the shapes of the vectors in the original page; so, this is a text-less page containing only the visual information shapes that will be subsequently used during the detection.

It is to note that two images are useful as, by using only the former, the text may be detected by the computer vision heuristics, bringing noise into the detection. By applying them on the latter, it is assured that only actual geometric shapes will be detected.



(a) **(b)**

Figure 3.3: Example of couple of raster-type pages

3.1.2.2 Interested string search

Remembering the specific application in the sustainability realm, in every report the final output of the materiality analysis is portrayed under the header "materiality matrix"; obviously, this string depends on the document's native language, so in order to automatize the research, the developed application allows the user to pass a three-characters argument that identifies the text language.
During this first step, with the help of the `PyMuPDF` library – described in section **(3.1.4)** –, the script simply searches through the document file

for every page where this string is present, marking them. As to be seen in the following section, this is useful to understand whether or not apply the heuristic that has been specifically set up for the materiality matrices detection.

3.1.2.3 Visual data detection

This is the most crucial passage of the entire process, as it actually consists into detect and retrieve the unstructured visual data.
Let's consider a couple of page-images (the ones described in **(3.1.1)**), for simplicity they will be referred with the couple (I_{PT}, I_{SO}) with I_{PT} being the plain-text page-image and I_{SO} being the shape-only page-image; two different heuristics are followed depending from the passage described in section **(3.1.2.2)**, and so depending on if the page has been marked as one containing a materiality matrix or not; both the heuristics are based on the application of different variants of the *Hough transform* algorithm **(2.1)**.

Materiality matrix presence If a materiality matrix is detected (by the OCR routine reading the page's header), an heuristic based on the *line Hough transform* **(2.1.1)** is applied on the I_{SO} image, after applying the *Canny edge detector*; this, because a two-dimensional-like plot is more likely to be detected by an algorithm based on line detection for the prevalence of rectangular shapes.
Precisely, through the *Hough transform* a set of vertices delimiting the plot can be found; obviously these are located in the same coordinates of the *Hough* lines intersections.

For completeness of information, the following parameters have been used during the implementation of the proposed solution:

- ***Canny***:
  - `threshold1`: 50 (first threshold for the hysteresis procedure);
  - `threshold2`: 150 (second threshold for the hysteresis procedure);
  - `apertureSize`: 3 (aperture size for the *Sobel* operator).
- ***Line Hough transform***:
  - `rho`: 1 (the resolution of the parameter ρ of the *Hough transform*, in pixels);
  - `theta`: $\frac{\pi}{180}$ (the resolution of the parameter θ of the *Hough transform*, in radians);

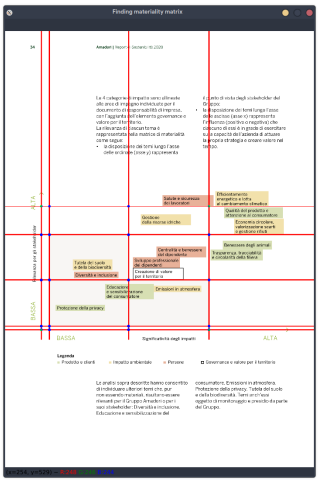
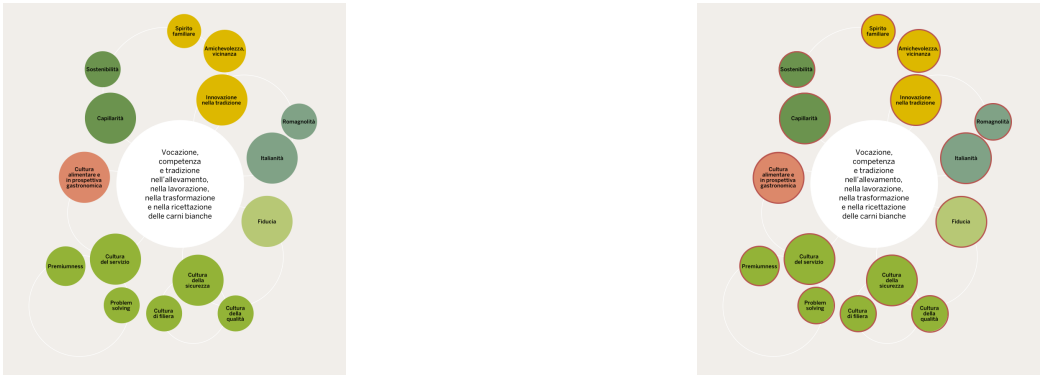


Figure 3.4: Example of HT-based materiality matrix detection

– `threshold`: 700 (the minimum number of intersections to detect a line).

Materiality matrix absence (general case) In the other cases, two different heuristics can be followed depending from the presence of rectangular shapes (such as squared infographics or tables) or circular ones. In the former case, the *line Hough transform* is applied just like has been described in the previous paragraph; in the latter case, if circular shapes are detected, an approach based on the *circle Hough transform* **(2.1.2)** is followed.
This particular solution has been set up in case of presence of circular shapes, which can be identified with generic infographics or pie/donut charts.



(a) **(b)**

Figure 3.5: Example of circle HT-based detection

For completeness of information, the following parameters have been used:

- ***Circle Hough transform***:
  - `dp`: 1 (the inverse ratio of resolution);
  - `minDist`: 150 (minimum distance between detected centers);
  - `param1`: 100 (upper threshold for the internal *Canny edge detector*);
  - `param2`: 51 (threshold for center detection);
  - `minRadius`: 100 (minimum radius to be detected);
  - `maxRadius`: 1000 (maximum radius to be detected).
- ***Line Hough transform***:
  - `threshold1`: 50 (first threshold for the hysteresis procedure);
  - `threshold2`: 150 (second threshold for the hysteresis procedure);
  - `apertureSize`: 3 (aperture size for the *Sobel* operator).

3.1.2.4 Export

Finally, if the detected visual data is located in the feasible area of the page (as described in **(3.1.1)**), it is considered as an interesting region to be extracted by cropping the I_{PT} image. At the same time is executed another routine that, instead, collects statistics on which data were collected and extracted correctly, by detecting the presence of likely paragraphs (this will be discussed in **(3.1.2.5)**). Depending from the approach, the crop is executed in a different manner:

Line *Hough* transform The (x, y) coordinates of the detected *Hough* lines intersection in I_{SO} are then memorized and used as delimiter points to crop the I_{PT} image, in order to obtain the final plot.

Circle *Hough* transform Let's take a look at **Figure 3.6**. The middle-points of the furthest four circles in I_{OS} (red vertices) are localized and then used to delimiter a rectangle (purple rectangle) through them. Following, the radius is computed for each one of them, and depending from the circle's position on the four cardinal points, the radius is distinguished as r_W (horizontally, for width) and as r_H (vertically, for height); the maximum for both of them $R_i > r_i$ $(i = W, H)$ is defined as the distance between the smaller

rectangle and a bigger one (the brown rectangle), which delimits the area to be cropped in the I_{PT} image.

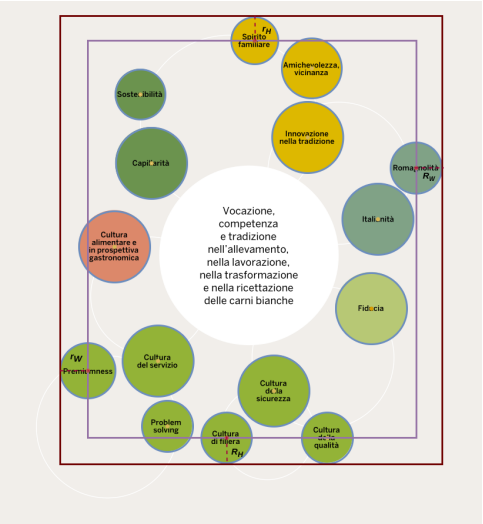


Figure 3.6: Circle *Hough transform* crop

3.1.2.5 Other possible interventions

Assumed that what has been discussed in the previous sections it's based on heuristics, obviously the exporting process cannot always give good results; in fact, the variety of documents and styles makes the tasks so difficult that the heuristics always give a result with a certain variable error consisting in images containing purposeless text surrounding the image (paragraphs under/over the interesting graphics) or just bad-cropped images.
In order to give the user the ability to correct this error, the application has been also provided of two procedures, a semi-automatic and a manual procedure, that may be used to have a better output.

These two procedures can be applied by using the appropriate parameter when executing the software on the input document.

Semi-automatic procedure After having checked that an image (or a set of them) have not been correctly extracted, the user can execute the application in "paragraph removal mode", consisting in a two passes heuristics aimed at removing the text paragraph surrounding the interested image(s).

- In the first place, the heuristics discussed in the previous sections is performed. Simultaneously, the actual presence of a likely paragraph

is checked.

- If the verification is successful, the following procedure is performed:
  - firstly, the page image is thresholded in order to maintain only plots and remove the text;
  - next, a 5×5-kernel dilatation is performed, followed by an opening characterised by the same kernel;
  - following, the negative image is found starting from the output of the previous step;
  - the non-zero points of the final image are found; these correspond with the graphics to be cropped;
  - the bounding-edges of the region are found and, after adding a padding, the corresponding area in the I_{SO} image is cropped to obtain the final result.

Manual procedure After having checked that an image (or a set of) have not been correctly extracted, the user is also able to execute the application in a "manual extraction mode", which consists in detecting the pages containing a plot or an image, and then manually cropping them through the help of the `roi` function from the `opencv-python` library described in section **(3.1.4)**.

3.1.3 Table extraction process

Although the visual data extraction focuses also on tabular shapes, being them rectangular or squared graphics, another approach may be useful in these specific situation. So, in order to have actual alphanumeric data extracted directly from these table, a different approach has been set up.

It has to be noted that more than one kind of table can be found in sustainability reports, some of them including general purpose informations (like statistics and parameters about the involved company) some others including the so-called *Global Reporting Initiative* (GRI) indices, referring to those "good practices" the company has been following during the year in order to increase its sustainability.
The tabular data extraction is performed through the following heuristic:

- firstly, each page is checked for the presence of a generic table; if it's present:

  - if it's a GRI table, it is extracted onto an opportune directory as a `CSV` file; then, through a regular expression, every GRI record is extracted. This is done to match all the possible patterns of strings included in the GRI records.
    More in particular, through the regex
    `(.*?)(.(\s(\D|\S)\s)|\S?)(.\d?)`
    strings like the following ones can be detected:
    * `GRI 104;`
    * `GRI104;`
    * `GRI-104;`
    * `GRI - 104;`
    * `GRI - 104 - 10`
    * `GRI 104 - 10`
    * `GRI104 - 10`
    * `G 4`
    * `G4`
    * `G-4`
    * `G - 4`
    * `G 4 - 10`
    * `G - 4 - 10`
  - if it's not a GRI table, it is just exported onto another opportune directory as a `CSV` file.

- if no table is found, the process is repeated on the next page.

3.1.4 Libraries

In this section will be listed and described the libraries used to develop the this specific application. As has already been stated, the resulting extraction software has been written in the *Python* programming language.

3.1.4.1 Third-party libraries

The following imported libraries are all available in the *PyPi* repository.

- `fitz` – Also known as *PyMuPDF*, it is a *Python* library for data extraction, analysis, conversion and manipulation of documents; its supported document formats include `PDF`, `XPS`, `EPUB`, `MOBI`, `FB2`, `CBZ`, `SVG` and various raster types of images. More informations about this library can be found at `https://pymupdf.readthedocs.io/en/latest/`.

- `pdfplumber` – Another library for `PDF`-format documents data extraction; more specifically, it is used for the table extraction process. More informations about this library can be found via the official GitHub repository: `https://github.com/jsvine/pdfplumber`.

- `opencv-python` – Pre-built CPU-only *OpenCV* packages for *Python*. *OpenCV* (***Open*** *Source* ***C****omputer* ***V****ision Library*) is an open source (Apache 2 License) computer vision and machine learning software library. This library contains more than 2500 algorithms, from both the classic and the state-of-the-art realms. It was used in order to apply the described computer vision methods during the extraction. More informations about this library can be found at `https://opencv.org/`.

- `pytesseract` – *Python-tesseract* is a wrapper for Google's *Tesseract-OCR Engine*. More informations about this library can be found at `https://pypi.org/project/pytesseract/`.

- `numpy` – *NumPy* is an open source project that enables numerical computing with Python and offers useful calculation functions compatible with different data structures. More informations about this library can be found at `https://numpy.org/`.

- `pandas` – An open source library for data visualization and analysis; it was used to build the dataframes containing the OCR data. More informations about this library can be found at `https://pandas.pydata.org/`.

3.1.4.2 From-scratch libraries

Following are listed and described the built libraries that envelop the heuristics that have been discussed in the previous sections.

- `document_page.py` – It contains the object class `DocumentPage`, describing a `PDF`-format file document page as seen in section **(3.1.1)** and containing all its informations (text, index number etc.).

- `languages.py` – It contains the dictionary `LANGUAGE_DICT` defining the word "materiality matrix" in different languages, used during the research of this string as header, as seen in section **(3.1.2.2)**.

- `plot_extractor.py` – It contains the object class `PDFToImage`, used for extraction of images and plots from the documents, as seen in section **(3.1.2)**.

- `tables_extractor.py` – It contains the object class `TableToCSV`, for the extraction of tables (by using the library `pdfplumber`) into `CSV`-format files, as seen in **(3.1.3)**.

3.2 Materiality matrix data extraction

As seen in the previous chapters, the materiality matrix is a dominant object obtained at the end of the entire analysis, through which the viewer is able to understand which are the organization's prime sustainability objectives; however, this plot has often different forms and structures, as well as being missing of exact quantitative data.

In order to have further comprehension of these plots, and gain actual quantitative data (which could be then exported in `CSV` format files and then imported in database instances to make machine learning and deep learning operations), has been build a *Python* software that, using different heuristics depending on the input matrix, calls the *PyTesseract* library – described in section **(3.1.4)** – and the *OpenCV* libraries, allowing the data extraction.

3.2.1 Format of the involved matrices

Because of the lack of an actual standard, amounts of formats can be distinguished: some materiality matrices are characterized by boxes, other ones by icons; some of them use a legend and some other not; some ones have labels inside the plot, other of them are instead outside.
This scarceness of "shape" constancy is another reason why materiality matrices are so difficult to be really understandable.

At the time of writing this thesis, the system is able to gain informations from the following types of materiality matrices:

- ***Box-Type***: bi-dimensional Cartesian plots characterised by labelled colored rectangles, containing texts describing the materiality themes, and sometimes followed by an outer legend grouping the macro-themes;

- ***Blob-Type***: bi-dimensional Cartesian plots characterised by blobs of variable size; here, the label is often presented in text (sometimes contained in a rectangle) connected to the nearest blob. They are often followed by a legend, as the previous ones.

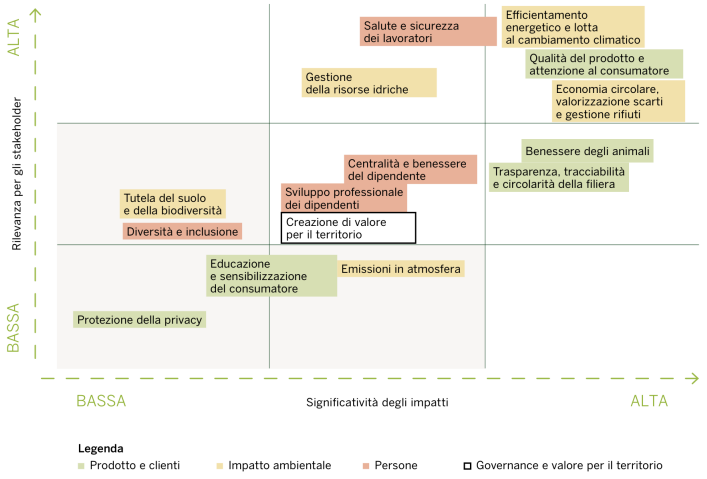


Figure 3.7: Example of *Box-Type* materiality matrix

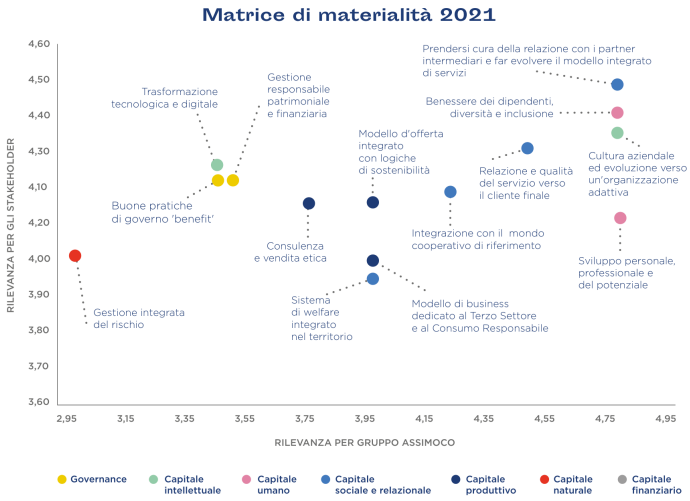


Figure 3.8: Example of *Blob-Type* materiality matrix

3.2.2 Data extraction process

The entire process is composed by four consecutive steps:

1. **plot/legend subdivision**: the input image, containing the entire materiality matrix with its legend, is divided into two sections: the plot and the legend;

2. **plot interpretation**: the plot section is analyzed and interpreted using computer vision heuristics: through this step, actual numeric data

is acquired;

3. **legend interpretation**: as for the previous step, the legend section is analyzed and contextualized with the previously acquired data;

4. **final export**: the data acquired in the previous steps is put together in a dataframe, and then exported into a comma-separated-value (`CSV`) file.

In the next sections, the process will be more accurately described for both the involved formats of plots.

3.2.2.1 Plot/Legend subdivision

As previously stated, whether the format of the materiality matrix, it is often made up by a plot section – which is the actual two-dimension chart explaining the materiality themes – and legend section – which, instead, groups these themes in bigger categories, called macro-themes.
Therefore, when a legend is actually present in an input image, it is firstly necessary to separate these two sections in order to do a further analysis of the two, through Computer Vision algorithms.

Obviously, the process has to be distinguished depending from the previously identified formats.

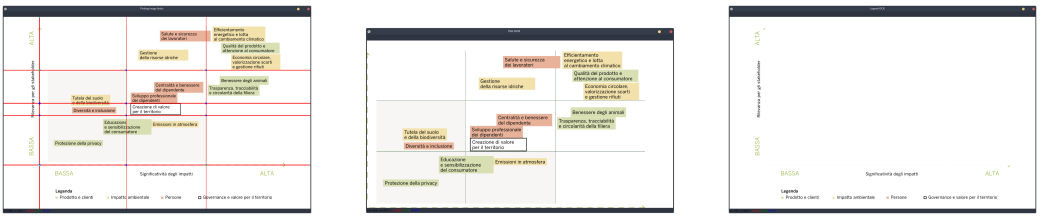
Box-Type In order to split the input image in two different sections, it is required to find all the "blocks" composing the plot, and so to find their delimiters.
Starting from the input matrix, a binary image is obtained through a RGB-to-grayspace conversion followed by a threshold operation: this makes possible to have an image containing only the major "shapes", on which will then be applied some computer vision algorithms.
Following, firstly is applied the *Canny algorithm* in order to detect the basic **edges** of the image, and then the *Hough transform* **(2.1)** is used to find all the lines passing through these edges. For completeness of information, the following parameters have been used:

- ***Canny***:
  - `threshold1`: 50 (first threshold for the hysteresis procedure);
  - `threshold2`: 150 (second threshold for the hysteresis procedure);

  - `apertureSize`: 3 (aperture size for the *Sobel* operator).

- ***Hough transform***:
  - `rho`: 1 (the resolution of the parameter ρ of the *Hough transform*, in pixels);
  - `theta`: $\frac{\pi}{180}$ (the resolution of the parameter θ of the *Hough transform*, in radians);
  - `threshold`: 450 (the minimum number of intersections to detect a line).

As stated before, finding the blocks composing the image means finding the vertices limiting them; these points are defined as the intersection between the lines detected through the *Hough transform* (**Figure 3.9a**).
Finally, knowing that the majority of the materiality matrices are composed by 3×3 blocks, a scale ratio by $1 : 3$ is used so as to find the remaining three limiting vertices including the plot section (respectively at the positions top-left, bottom-left and bottom-right); once this section is identified by its delimiting vertices, the original input image is divided into the two aforementioned sections (**Figure 3.9b** and **Figure 3.9c**).



(a) **(b)** **(c)**

Figure 3.9: Passes of the *Box-Type* matrices subdivision

Blob-Type Again, it is first of all indispensable to binarize the input image: as before, this is done by making a RGB-to-grayspace color-space conversion followed by a thresholding operation; however, in this format there is a predominance of circular shapes, more than rectangular ones, an approach based on a *blob detector* **(2.2)** has been chosen.
The heuristic is made up by the following consecutive steps:

- at first, using the detector, the software finds all the blobs (circular shapes) that have the same coordinates, disposed both in horizontal and in vertical (**Figure 3.10a**);

- next, to distinguish the legend's blob from the actual plot's blob (representing the values in (x, y) coordinates), the software counts the number of blobs that share the same coordinates (**Figure 3.10b**);
- finally, it detects the first and the last blob composing the legend, and using them as delimiters, it "crops" the legend away from the input image (**Figure 3.10c**).

(a) **(b)** **(c)**

Figure 3.10: Passes of the *Blob-Type* matrices subdivision

3.2.2.2 Plot interpretation

The interpretation is obviously the main passage of the plot processing; requiring notions about the shapes geometry and the ability to "read" its contained content (this is easily done using OCR libraries), it is divided into the about to be mentioned subsequent steps:

1. shapes detection;
2. text detection;
3. legend link.

Following they will be accurately described, distinguishing the aforementioned formats of plot.

Shapes detection In the first place, the input plot section is binarized (again, but this time a different threshold is used so to obtain exclusively the shapes of the elements portrayed in the image); next, a 2×2-kernel dilatation is done in order remove the thin lines that may be in the image (this to avoid that some of the shapes crossing these lines would not be detected).
The actual corner extraction is following described, depending from the plot's format.

- ***Box-Type*** – Detecting shapes in this specific context, actually consists in finding the corners forming the rectangular labels contained in the plot. Firstly, after applying a threshold to remove the text, a 3×3-kernel dilatation is performed. Hence, all contours can be easily detected through a contour approximation algorithm, allowing to obtain the vertices of every single rectangle in the plot, the most important of which are the one on the top-left and the on the bottom-right. From just these two vertices, every rectangle could easily be described.
  However, a more complex case has lead to the research of another solution. Some plots belonging from this class, often presents label being so near that the boxes including them result "joined". Therefore, to detect them, is necessary to distinguish four-edges shapes from six-edges one:

  – shapes having **four edges** are simple rectangles on which the aforementioned method could be used;
  – shapes having **six edges** are complex polygons made up by two rectangles.

  Remembering that the origin of this coordinates system is located in the top-left, this implies the detection of the following vertices:

  – the point having **minimum** x coordinate (being the furthest on the left);
  – the point having the **maximum** x coordinate (being the furthest on the right);
  – the point having the **minimum** y coordinate (being the furthest on the top);
  – the point having the **maximum** y coordinate (being the furthest on the bottom).

  Once found all of them, it's easy to find the actual position of every rectangle by calculating their middle-point, as can be seen in the following image.

  Given A_i and D_i for the i-th rectangle composing the complex polygon, the middle-point R_i can be found as following:

$$A_i \equiv (x_{A_i}, y_{A_i}) \quad D_i \equiv (x_{D_i}, y_{D_i}) \tag{3.1}$$

$$R_i \equiv \left(\frac{x_{A_i} + x_{D_i}}{2}, \frac{y_{A_i} + y_{D_i}}{2} \right) \tag{3.2}$$

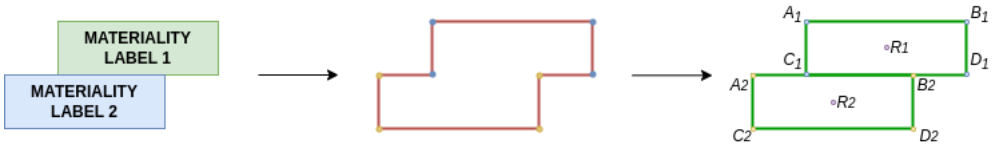


Figure 3.11: Steps of the rectangular shapes detection

Having identified the major vertices of the rectangles forming the complex polygon, their description is then complete.

- ***Blob-Type*** – Here the procedure is definitely simpler than in the previous case: after thresholding – in order to remove the text on the input plot – and applying a 3×3-kernel dilatation – to remove the thin lines – a blob detector is used. Through it, the entirety of circular shapes is detected and localized, so to have a complete description of them.

As the detection is completed, in both the cases the shape description, which will be from now on noted as `LabelBox` in *Box-Type* plots and `BlobBox` in *Blob-Type* plots (these are the name of the built class objects that describe the shapes), is memorized in a data structure along with the color of the shape itself; this, in fact, will be later used to contextualize the gathered data with what will be eventually detected in the legend section (if present).

Text detection Another threshold operation is done on the plot section in input, but this time in manner to leave only the text and ignore the shapes. During this pass, two different cases can be distinguished:

- **black text on colored images**: the binarized image has exclusively black text, without any background (as it is discarded during the thresholding);
- **white text on colored images**: the binarized image has white text on black boxes.

As OCRs perform better on black-texted images, in the latter case the resulting binarized image is converted to negative, so to proceed like in the former one.
Then, the `Tesseract-OCR` **(2.3.2)** routine (called via the `pyTesseract` wrapper library) is able to retrieve a dictionary containing every single word in the image, its bounding box geometrical data and its pixel-coordinates. All these informations are then used to instantiate `TextBox` class objects for every detected word.

However, even in this step, different pipelines have to be distinguished depending from the plot's format.

- ***Box-Type*** – The bounding boxes and the respective coordinates are used to detect which word is contained in which colored box, and thus in which `LabelBox`, forming then the label's complete string.
- ***Blob-Type*** – Every `TextBox` is put into a `LabelBoxColorless` (the name of the class-object defining the bounding boxes of those not colored labels being in *Blob-Type* plots) according to the following **heuristic**. Let t_x and t_y two thresholding values:
  - if the current word has a distance $d_x < t_x$ from next one, then they are on the **same row**;
  - else, the next word is in the **following row** if the distance between the first word and the first row has a distance $d_y < t_y$.

Finally, each circular shape detected in the previous step is joined to the nearest `LabelBoxColorless` (using the **euclidean distance** between the center of the blob and the center of the `LabelBoxColorless`).

Legend link Meanwhile, for every rectangle or blob in the plot, the color data is memorized both in RGB and HSV format (as it will be used in the following step) and then the `LabelBoxes` or `BlobBoxes` data is finally given as output.

3.2.2.3 Legend interpretation

If actually present, this passage is performed on the legend-section acquired during the first step.
At first, by using again the OCR, every information about text is acquired in the same manner as the previous passage. It should be noted that the legend's elements appear in the following manner:

Figure 3.12: Example containing both rectangular and circular colored shapes

So, converting the legend image in HSV color-space format, and then taking the non-zero point for every color obtained during the plot extraction

step, it is possible to detect the colored shape's position.
Next the legend is processed: knowing that the first `TextBox` of each legend-string is next to the color shape, the *euclidean distance* between them is minimum; to avoid errors, a threshold is used. Given this minimum distance, it is known that the first word of the label is owned by that label of the legend: so, it is created a `LegendBox` object identified by the position of the colored shape.

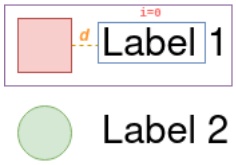


Figure 3.13: Caption

3.2.2.4 Final export

During the exporting pass, `LabelBoxes` and `LegendBoxes` are finally given as input to an `Exporter` class object: if there is no actual legend in the input image, the legend data is set as `None` and then discarded; if instead it is present, the following operations are done.

Normalization Given the middle-point of every `LabelBox`, its coordinates are normalized in the $[0, 300]$ range, in order to have – both horizontally and vertically – exactly 100 units for each block composing the plot.

CSV export The collected data is gathered in a `pandas` dataframe and then exported in a `CSV` file on the disk.

	Label	Macrotheme	Group Rel	Stake Rel	Rank Group	Rank Stake	Rank Absolute	Alignment
0	E innovazione di prodotto	Unknown	300	0	10	0	10	100
1	Formazione e sviluppo dei dipendenti diversit ed inclusione	Integrit	91.51	65.36	4.44	1.11	4.44	0
2	Consumo di acqua	Soddisfazione del cliente	0	73.52	0	2.22	0	17.3
3	Consumo energetico impatti sociali	Integrit	52.32	94.86	2.22	3.33	2.22	5.98
4	Ed emissioni di	Soddisfazione del cliente	12.93	101.66	1.11	4.44	1.11	22.85
5	Produzione e gestione	Sociale	61	136.61	3.33	5.56	3.33	18.06
6	Rifiuti	Integrit	106.95	147.5	5.56	6.67	5.56	5.26
7	Catena di fornitura sostenibile	Artigianalit e innovazione di prodotto	192.08	226.48	7.78	7.78	7.78	3.01
8	Diritti umani	Sociale	171.24	260.51	6.67	8.89	6.67	23.05
9	Etica aziendale e compliance	Governance e compliance	195.95	300	8.89	10	8.89	28.45

Figure 3.14: Materiality data extracted in `pandas` dataframe

PNG export (Open-format plot) The `pandas` dataframe data is used to generate a sort of "standardised" plot using the `matplotlib` library, and finally saved on a local directory.

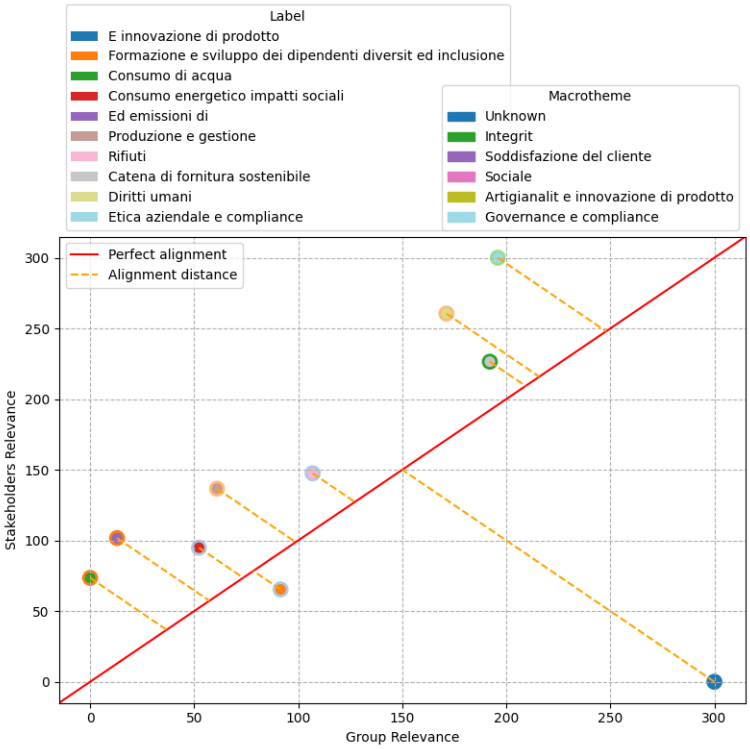


Figure 3.15: Materiality matrix converted in an open-format plot

3.2.3 Libraries

In this section will be listed and described the libraries used to develop the materiality matrix interpretation software. More informations about libraries that have already been used during the development of the visual data extraction process can be found at section **(3.1.4)**.

3.2.3.1 Third-party libraries

The following imported libraries are all available in the *PyPi* repository.

- `opencv-python` – It was used in order to apply the described computer vision methods during the extraction.
- `pytesseract` – It was used to read the materiality matrices textual data.
- `numpy` – Used for different mathematical functions.

- `pandas` – It was used to build the dataframes containing the converted data.
- `matplotlib` – A library to visualize numeric data in plot form. It was used to create the "standardised"-format plot. More informations about this library can be found at `https://matplotlib.org/`.

3.2.3.2 From-scratch libraries

Following are listed and described the built libraries that envelop the heuristics that have been discussed in the previous sections.

- `plot_cropper.py` – It contains the object classes – respectively, `Cropper` for *Box-Type* and `BlobCropper` for *Blob-Type* plots – involved in the plot/legend subdivision.
- `plot_elements.py` – It contains all the object classes defining the shapes that could be found in a generic plot. More in detail, it envelopes the following object classes:
  - `Box`, describing a generic rectangle;
  - `TextBox`, describing the bounding box of a single word;
  - `LabelBox`, describing a rectangle that contains an entire label and featuring a color;
  - `LabelBoxColorless`, describing a colorless rectangle (more like a simple bounding box) that contains an entire label;
  - `Blob`, describing a circular shape;
  - `BlobBox`, describing circular shapes attached to textual data and featuring a color.
  - `LegendBox`, describing the blocks that contain a legend label, characterized by its own color.
- `ocr.py` – This library recalls the *Google's TesseractOCR* routine, and defines different kinds of object classes depending from the part of the input image to be analyzed and from where extract data.
  - `OCR`, defining the Optical Character Recognition; it externally calls a `TesseractOCR` routine using the *pytesseract* library;
  - `PlotOCR`, defining the heuristics used depending from the different type of plots (it's an `OCR` subclass);

  - `PlotOCR_Box`, useful to detect the textual elements of the image's plot section in *Box-Type* plots (`PlotOCR` subclass);
  - `PlotOCR_Blob`, useful to detect the textual elements of the image's plot section in *Blob-Type* plots (`PlotOCR` subclass);
  - `LegendOCR`, useful to detect the textual elements of the image's legend section (`OCR` subclass).

- `exporter.py` – This library defines the object class `Exporter` committed to export the acquired data in both `CSV` and `PNG` (plot) format.

4
Results and conclusions

This chapter involves the test of the two applications on a sample of 100 different sustainability reports released from several Italian companies and randomly chosen from a bigger dataset.
The tests (divided in sections for each developed application) have been made on a self-built *Docker* container in a desktop machine having the following specifications:

- **CPU**: Intel(R) Core(TM) i5-6500 CPU @ 3.20GHz
- **RAM**: 23.4 GB
- **OS**: `Debian GNU/Linux 12 (bookworm)`
- **Kernel**: `Linux 6.1.0-12-amd64`
- ***Docker* Specifications**:
  - **Docker Version**: `Docker 24.0.6, build ed223bc`
  - **Container OS**: `Debian GNU/Linux 12 (bookworm)`
  - **Container `Tesseract-OCR` version**: `tesseract 5.3.0`
  - **Container *Python* version**: `Python 3.10.13`

4.1 Extraction from documents

Following are portrayed two tables showing the results of the extraction from 100 documents. More in specific are showed name of the **company** releasing

the report in which **year**, **total amount of extracted images** with details on the amount of **ambiguous** ones and **materiality matrices**, and **total amount of extracted tables** with details on the amount of those containing **GRI** indices.
The documents are ordered by execution during this first part of the test, which took approximately 12 hours (the majority of the considered documents have about 200 pages, with only some of them having about 300).

N	Company	Year	Tot. Images	Mat. Matrices	Ambig. Images	Tot. Tables	GRI Tables
1	Banco BPM	2021	182	2	50	42	6
2	CIR S.p.A.	2017	99	0	22	83	5
3	La Cassa di Ravenna S.p.A.	2018	78	11	3	11	1
4	Gefran S.p.A	2019	350	8	131	104	1
5	Vittoria Assicurazioni	2017	92	2	8	129	5
6	Banca Carige S.p.A.	2021	168	3	8	132	16
7	Finmeccanica	2013	182	1	45	182	13
8	Banca Sella S.p.A	2021	132	2	23	60	9
9	Emak	2020	76	1	9	84	80
10	Sogefi Group	2017	133	3	28	87	6
11	SERI Industrial Group	2019	77	2	14	87	5
12	Fincantieri	2019	109	1	52	45	7
13	Sogefi Group	2018	151	3	47	134	21
14	Gruppo CVA	2018	53	1	9	59	41
15	BPER Banca S.p.A.	2019	164	3	15	113	48
16	ASTM	2016	204	1	27	137	14
17	Volksbank	2021	80	3	20	65	9
18	BPER Banca S.p.A.	2020	192	1	15	141	51
19	GEDI Gruppo Editoriale	2018	84	3	15	41	27
20	Autogrill S.p.A.	2016	128	3	10	47	4
21	CIR S.p.A.	2019	111	0	36	133	24
22	Gruppo Dolomiti Energia	2017	102	2	28	86	36
23	Vittoria Assicurazioni	2021	132	4	45	186	26
24	Rai Way	2018	109	5	53	75	40
25	Esselunga	2021	167	2	85	93	31
26	Crédit Agricole	2020	128	3	35	79	79
27	SARAS	2020	172	4	34	100	15
28	Reply	2019	57	2	8	16	5
29	Autostrade per l'Italia	2021	138	2	39	98	54
30	illimity Bank	2020	124	5	18	127	68
31	Gruppo Aeffe	2021	64	2	3	48	9
32	DiaSorim	2021	100	2	10	98	44
33	La Cassa di Ravenna S.p.A.	2019	116	5	7	142	63
34	Crédit Agricole	2019	107	3	8	53	39
35	Feralpi S.p.A.	2020	192	4	188	277	31
36	Mondadori S.p.A.	2018	73	3	5	76	2
37	Infrastrutture Wireless Italiane S.p.A.	2019	38	2	17	45	15
38	2iReteGas	2021	160	3	40	110	41
39	Salvatore Ferragamo S.p.A.	2019	57	1	1	26	2
40	Mondadori S.p.A.	2014	127	1	16	85	21
41	Crédit Agricole	2019	124	1	12	83	83
42	Safilo	2019	45	3	11	51	40
43	Rai Way	2017	109	5	25	199	93
44	MARR S.p.A.	2020	67	2	40	72	26
45	Enel S.p.A.	2012	191	4	15	388	286
46	NEXI S.p.A.	2020	130	10	12	20	10
47	Salvatore Ferragamo S.p.A.	2018	55	0	1	26	9
48	La Doria S.p.A.	2021	118	2	44	145	80
49	La Valsabbina S.p.A.	2020	57	2	18	67	9
50	Helvetia Italia	2018	100	3	3	49	29

Table 4.1: Results on the first 50 documents

N	Company	Year	Tot. Images	Mat. Matrices	Ambig. Images	Tot. Tables	GRI Tables
51	TPER	2019	176	2	31	244	57
52	Vittoria Assicurazioni	2018	114	2	12	158	11
53	BiEsse Group	2019	77	2	11	172	56
54	Alperia	2020	179	2	9	386	208
55	Invitalia	2021	128	4	9	19	10
56	Autogrill S.p.A.	2014	139	7	9	17	14
57	Vittoria Assicurazioni	2020	124	3	29	179	29
58	Enav	2020	216	6	41	135	42
59	BPER Banca	2021	198	2	18	146	50
60	RCS Mediagroup S.p.A.	2020	106	3	32	73	36
61	Rai S.p.A.	2017	136	2	21	151	19
62	Pininfarina S.p.A.	2019	62	2	18	37	35
63	CiviBank	2019	73	3	42	117	76
64	La Doria	2017	96	1	42	117	41
65	Enav	2019	189	7	134	126	59
66	Banca Popolare di Sondrio	2020	136	6	33	101	42
67	Banca Generali	2015	126	2	24	205	12
68	Amadori S.p.A.	2021	132	1	26	73	45
69	Mondadori S.p.A.	2015	136	2	11	113	16
70	NEXI S.p.A.	2019	110	2	4	19	2
71	Salvatore Ferragamo S.p.A.	2016	51	0	3	26	1
72	La Cassa di Ravenna	2018	104	4	7	128	65
73	La Cassa di Ravenna	2020	125	5	7	138	67
74	GEDI Gruppo Editoriale S.p.A.	2019	47	1	15	108	55
75	Emak S.p.A.	2021	77	1	7	86	82
76	Saras S.p.A.	2017	98	3	8	95	10
77	La Doria S.p.A.	2019	114	1	43	125	65
78	Volksbank S.p.A.	2019	60	4	9	18	5
79	Banca Valsabbina S.p.A.	2021	64	2	4	97	9
80	Acsm Agam S.p.A.	2021	136	4	39	91	71
81	MARR S.p.A.	2021	71	2	54	62	9
82	Maire Tecnimont S.p.A.	2019	137	2	30	62	6
83	CEMBRE S.p.A.	2019	50	3	6	48	7
84	CIR S.p.A.	2018	111	3	29	121	24
85	CiviBank S.p.A.	2020	75	2	43	135	94
86	2iReteGas S.p.A.	2021	178	3	35	76	44
87	Amadori S.p.A.	2020	116	1	23	51	27
88	ITAS Assicurazioni S.p.A.	2020	108	1	40	152	28
89	Leonardo S.p.A.	2019	172	2	76	123	36
90	Invitalia S.p.A.	2020	359	2	114	343	78
91	Mondadori S.p.A.	2016	148	2	14	120	8
92	Esselunga S.p.A.	2019	252	1	42	131	25
93	Alperia S.p.A.	2018	91	3	17	144	136
94	La Valsabbina S.p.A.	2019	41	1	5	59	7
95	Civibank S.p.A.	2021	81	1	22	128	74
96	Banca Popolare Pugliese	2019	116	3	13	69	22
97	Fincantieri	2017	96	4	37	51	9
98	Crédit Agricole	2021	140	3	70	77	61
99	Assimoco S.p.A.	2021	196	2	62	101	15
100	Autogrill S.p.A.	2022	304	1	17	138	95

Table 4.2: Results on the last 50 documents

	Total	Materiality matrices	Ambiguous
Extracted Images	12326	2.17%	22.83%

Table 4.3: Overall statistics on image extraction

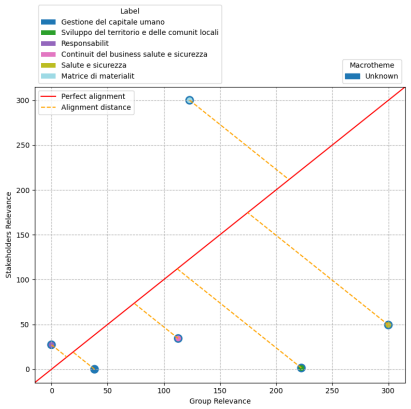
	Total	GRI
Extracted Tables	10403	36.57%

Table 4.4: Overall statistics on table extraction

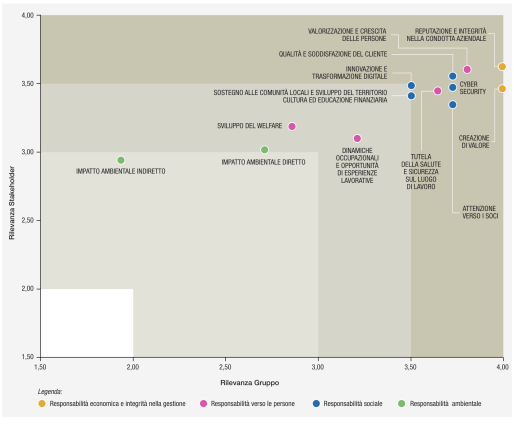
4.2 Materiality matrices interpretation

In this part of the test, which took about 30 minutes, 50 manually-cropped materiality matrices were involved; for the sake of brevity, only 20 of these are portrayed below.

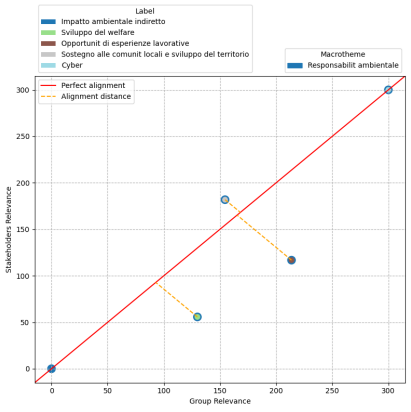
(a)



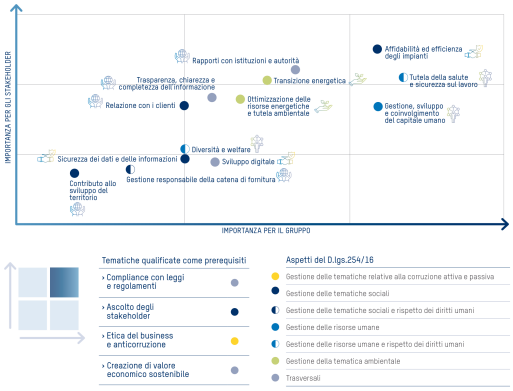
(b)



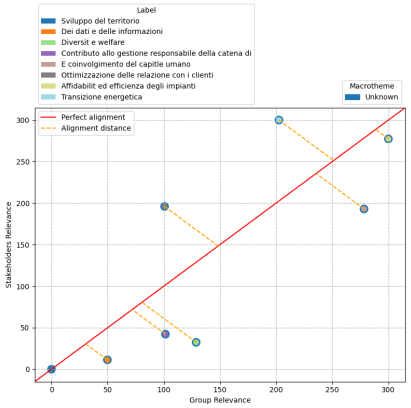
(a)



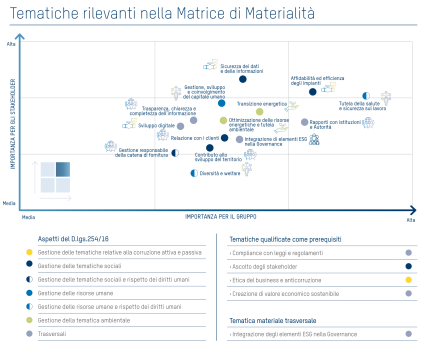
(b)


IMPORTANZA PER GLI STAKEHOLDER
IMPORTANZA PER IL GRUPPO
Rapporti con istituzioni e autorità
Affidabilità ed efficienza degli impianti
Trasparenza, chiarezza e completezza dell'informazione
Transizione energetica
Tutela della salute e sicurezza sul lavoro
Relazione con i clienti
Ottimizzazione delle risorse energetiche e tutela ambientale
Gestione, sviluppo e coinvolgimento del capitale umano
Diversità e welfare
Sicurezza dei dati e delle informazioni
Sviluppo digitale
Contributo allo sviluppo del territorio
Gestione responsabile della catena di fornitura
Tematiche qualificate come prerequisiti
› Compliance con leggi e regolamenti
› Ascolto degli stakeholder
› Etica del business e anticorruzione
› Creazione di valore economico sostenibile
Aspetti del D.lgs.254/16
Gestione delle tematiche relative alla corruzione attiva e passiva
Gestione delle tematiche sociali
Gestione delle tematiche sociali e rispetto dei diritti umani
Gestione delle risorse umane
Gestione delle risorse umane e rispetto dei diritti umani
Gestione della tematica ambientale
Trasversali

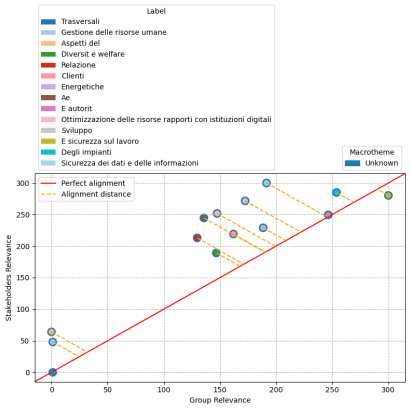
(a)


Label
Sviluppo del territorio
Dei dati e delle informazioni
Diversit e welfare
Contributo allo gestione responsabile della catena di
E coinvolgimento del capitle umano
Ottimizzazione delle relazione con i clienti
Affidabilit ed efficienza degli impianti
Transizione energetica
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

(b)


Tematiche rilevanti nella Matrice di Materialità
IMPORTANZA PER GLI STAKEHOLDER
IMPORTANZA PER IL GRUPPO
Alta
Media
Aspetti del D.lgs.254/16
Tematiche qualificate come prerequisiti
Tematica materiale trasversale

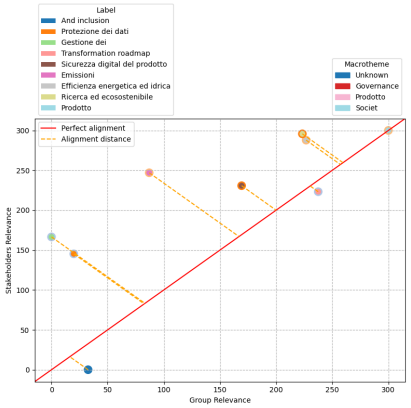
(a)


Label
Trasversali
Gestione delle risorse umane
Aspetti del
Diversit e welfare
Clienti
Relazione
Energetiche
Ae
E autorit
Ottimizzazione delle risorse rapporti con istituzioni digitali
Sviluppo
E sicurezza sul lavoro
Degli impianti
Sicurezza dei dati e delle informazioni
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

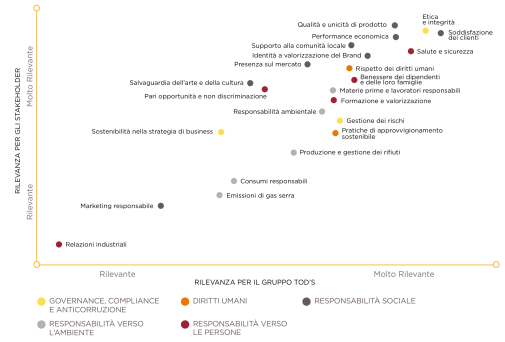
(b)


STAKEHOLDER ESTERNI
BREMBO
Rilevante
Molto rilevante
Ambiente
Società
Governance
Prodotto

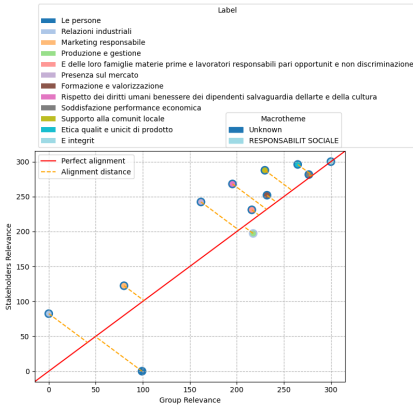
(a)


Label
And inclusion
Protezione dei dati
Gestione dei
Transformation roadmap
Sicurezza digital del prodotto
Emissioni
Efficienza energetica ed idrica
Ricerca ed ecosostenibile
Prodotto
Macrotheme
Unknown
Governance
Prodotto
Societ
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

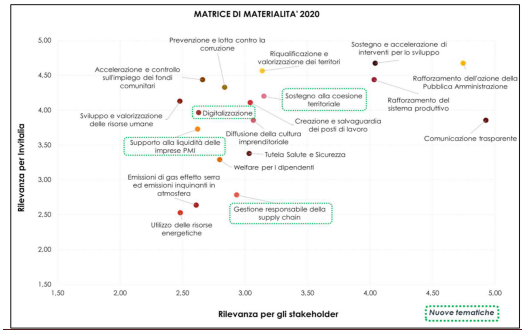
(b)


RILEVANZA PER GLI STAKEHOLDER
Molto Rilevante
Rilevante
Rilevante
Molto Rilevante
RILEVANZA PER IL GRUPPO TOD'S
Qualità e unicità di prodotto
Etica e integrità
Soddisfazione dei clienti
Performance economica
Supporto alla comunità locale
Identità e valorizzazione del Brand
Salute e sicurezza
Presenza sul mercato
Rispetto dei diritti umani
Benessere dei dipendenti e delle loro famiglie
Salvaguardia dell'arte e della cultura
Pari opportunità e non discriminazione
Materie prime e lavoratori responsabili
Formazione e valorizzazione
Responsabilità ambientale
Gestione dei rischi
Sostenibilità nella strategia di business
Pratiche di approvvigionamento sostenibile
Produzione e gestione dei rifiuti
Consumi responsabili
Emissioni di gas serra
Marketing responsabile
Relazioni industriali
GOVERNANCE, COMPLIANCE E ANTICORRUZIONE
DIRITTI UMANI
RESPONSABILITÀ SOCIALE
RESPONSABILITÀ VERSO L'AMBIENTE
RESPONSABILITÀ VERSO LE PERSONE

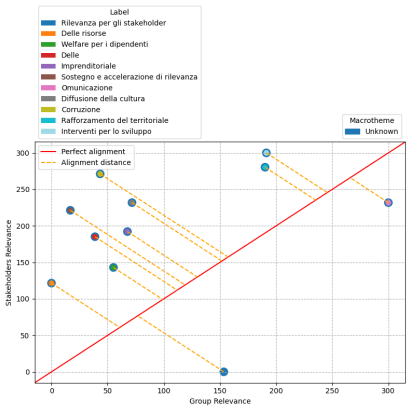
(a)


Label
Le persone
Relazioni industriali
Marketing responsabile
Produzione e gestione
E delle loro famiglie materie prime e lavoratori responsabili pari opportunit e non discriminazione
Presenza sul mercato
Formazione e valorizzazione
Rispetto dei diritti umani benessere dei dipendenti salvaguardia dellarte e della cultura
Soddisfazione performance economica
Supporto alla comunit locale
Etica qualit e unicit di prodotto
E integrit
Macrotheme
Unknown
RESPONSABILIT SOCIALE
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

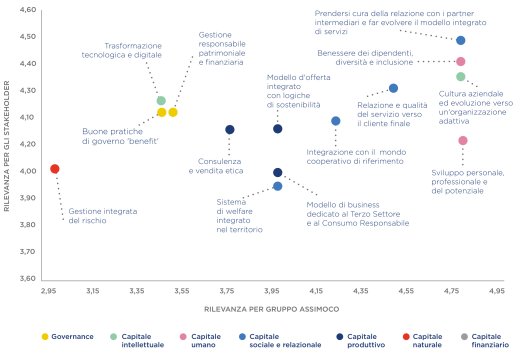
(b)


MATRICE DI MATERIALITA' 2020
Rilevanza per Invidia
Rilevanza per gli stakeholder
Prevenzione e lotta contro la corruzione
Riqualificazione e valorizzazione dei territori
Sostegno e accelerazione di interventi per lo sviluppo
Accelerazione e controllo sull'impiego dei fondi comunitari
Rafforzamento dell'azione della Pubblica Amministrazione
Sostegno alla coesione territoriale
Rafforzamento del sistema produttivo
Sviluppo e valorizzazione delle risorse umane
Digitalizzazione
Creazione e salvaguardia dei posti di lavoro
Comunicazione trasparente
Diffusione della cultura imprenditoriale
Supporto alla liquidità delle imprese PMI
Tutela Salute e Sicurezza
Welfare per i dipendenti
Emissioni di gas effetto serra ed emissioni inquinanti in atmosfera
Gestione responsabile della supply chain
Utilizzo delle risorse energetiche
Nuove tematiche

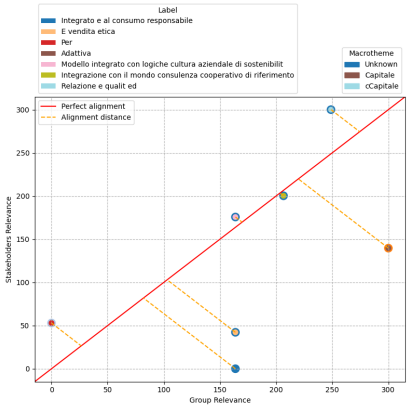
(a)


Label
Rilevanza per gli stakeholder
Delle risorse
Welfare per i dipendenti
Delle
Imprenditoriale
Sostegno e accelerazione di rilevanza
Omunicazione
Diffusione della cultura
Corruzione
Rafforzamento del territoriale
Interventi per lo sviluppo
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

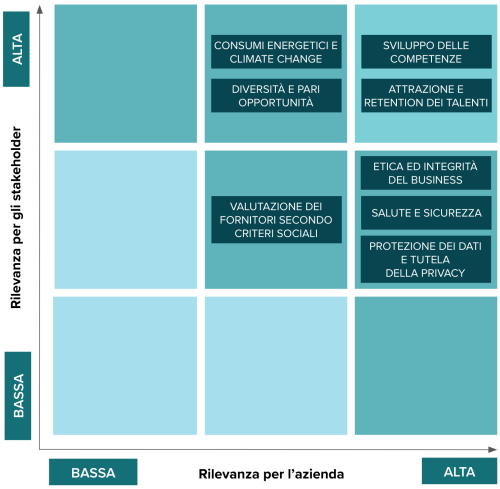
(b)


RILEVANZA PER GLI STAKEHOLDER
RILEVANZA PER GRUPPO ASSIMOCO
Governance
Capitale intellettuale
Capitale umano
Capitale sociale e relazionale
Capitale produttivo
Capitale naturale
Capitale finanziario
Trasformazione tecnologica e digitale
Gestione responsabile patrimoniale e finanziaria
Prendersi cura della relazione con i partner intermediari e far evolvere il modello integrato di servizi
Benessere dei dipendenti, diversità e inclusione
Modello d'offerta integrato con logiche di sostenibilità
Relazione e qualità del servizio verso il cliente finale
Cultura aziendale ed evoluzione verso un'organizzazione adattiva
Buone pratiche di governo "benefit"
Integrazione con il mondo cooperativo di riferimento
Consulenza e vendita etica
Sviluppo personale, professionale e del potenziale
Sistema di welfare integrato nel territorio
Modello di business dedicato al Terzo Settore e al Consumo Responsabile
Gestione integrata del rischio

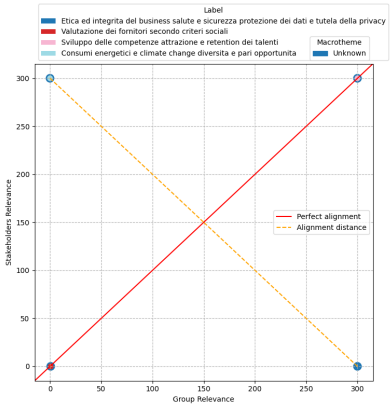
(a)


Label
Integrato e al consumo responsabile
E vendita etica
Per
Adattiva
Modello integrato con logiche cultura aziendale di sostenibilit
Integrazione con il mondo consulenza cooperativo di riferimento
Relazione e qualit ed
Macrotheme
Unknown
Capitale
rCapitale
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

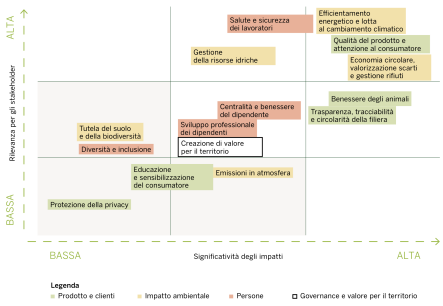
(b)


ALTA
Rilevanza per gli stakeholder
BASSA
CONSUMI ENERGETICI E CLIMATE CHANGE
SVILUPPO DELLE COMPETENZE
DIVERSITÀ E PARI OPPORTUNITÀ
ATTRAZIONE E RETENTION DEI TALENTI
VALUTAZIONE DEI FORNITORI SECONDO CRITERI SOCIALI
ETICA ED INTEGRITÀ DEL BUSINESS
SALUTE E SICUREZZA
PROTEZIONE DEI DATI E TUTELA DELLA PRIVACY
BASSA
Rilevanza per l'azienda
ALTA

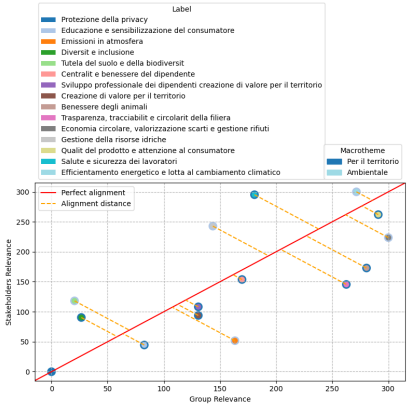
(a)


Label
Etica ed integrita del business salute e sicurezza protezione dei dati e tutela della privacy
Valutazione dei fornitori secondo criteri sociali
Sviluppo delle competenze attrazione e retention dei talenti
Consumi energetici e climate change diversita e pari opportunita
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

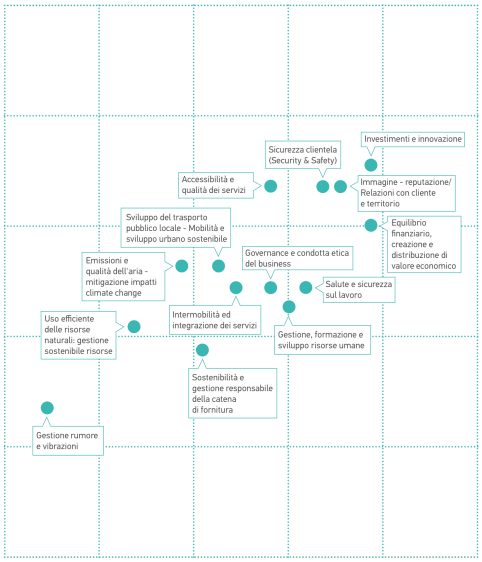
(b)


ALTA
Rilevanza per gli stakeholder
BASSA
Salute e sicurezza dei lavoratori
Efficientamento energetico e lotta al cambiamento climatico
Gestione della risorse idriche
Qualità del prodotto e attenzione al consumatore
Economia circolare, valorizzazione scarti e gestione rifiuti
Benessere degli animali
Centralità e benessere del dipendente
Trasparenza, tracciabilità e circolarità della filiera
Tutela del suolo e della biodiversità
Sviluppo professionale dei dipendenti
Creazione di valore per il territorio
Diversità e inclusione
Educazione e sensibilizzazione del consumatore
Emissioni in atmosfera
Protezione della privacy
BASSA
Significatività degli impatti
ALTA
Legenda
Prodotto e clienti
Impatto ambientale
Persone
Governance e valore per il territorio

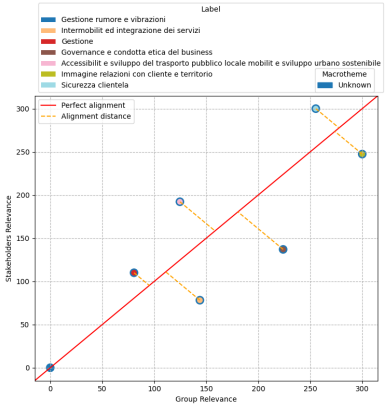
(a)


Label
Protezione della privacy
Educazione e sensibilizzazione del consumatore
Emissioni in atmosfera
Diversit e inclusione
Tutela del suolo e della biodiversit
Centralit e benessere del dipendente
Sviluppo professionale dei dipendenti creazione di valore per il territorio
Creazione di valore per il territorio
Benessere degli animali
Trasparenza, tracciabilit e circolarit della filiera
Economia circolare, valorizzazione scarti e gestione rifiuti
Gestione della risorse idriche
Qualit del prodotto e attenzione al consumatore
Salute e sicurezza dei lavoratori
Efficientamento energetico e lotta al cambiamento climatico
Macrotheme
Per il territorio
Ambientale
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

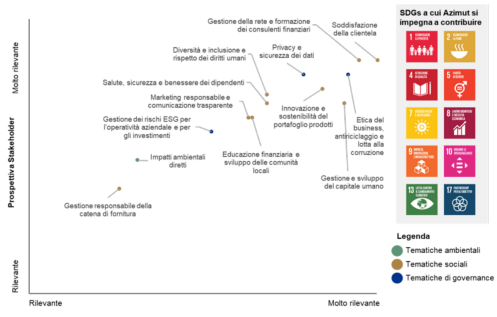
(b)


Investimenti e innovazione
Sicurezza clientela (Security & Safety)
Accessibilità e qualità dei servizi
Immagine - reputazione/ Relazioni con cliente e territorio
Sviluppo del trasporto pubblico locale - Mobilità e sviluppo urbano sostenibile
Equilibrio finanziario, creazione e distribuzione di valore economico
Emissioni e qualità dell'aria - mitigazione impatti climate change
Governance e condotta etica del business
Salute e sicurezza sul lavoro
Uso efficiente delle risorse naturali: gestione sostenibile risorse
Intermobilità ed integrazione dei servizi
Gestione, formazione e sviluppo risorse umane
Sostenibilità e gestione responsabile della catena di fornitura
Gestione rumore e vibrazioni

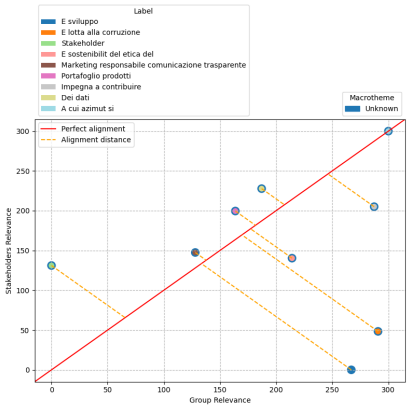
(a)


Label
Gestione rumore e vibrazioni
Intermobilit ed integrazione dei servizi
Gestione
Governance e condotta etica del business
Accessibilit e sviluppo del trasporto pubblico locale mobilit e sviluppo urbano sostenibile
Immagine relazioni con cliente e territorio
Sicurezza clientela
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

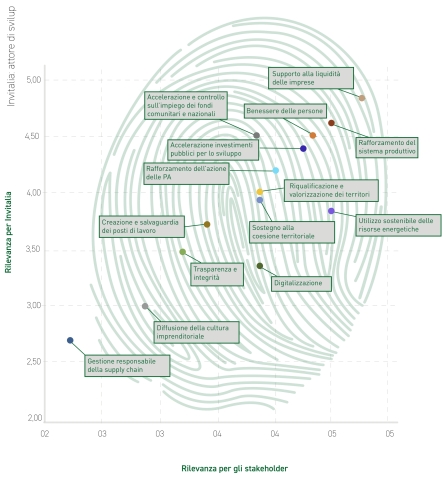
(b)


Prospettiva Stakeholder
Molto rilevante
Rilevante
Rilevante
Molto rilevante
Gestione della rete e formazione dei consulenti finanziari
Soddisfazione della clientela
Diversità e inclusione e rispetto dei diritti umani
Privacy e sicurezza dei dati
Salute, sicurezza e benessere dei dipendenti
Marketing responsabile e comunicazione trasparente
Innovazione e sostenibilità del portafoglio prodotti
Gestione dei rischi ESG per l'operatività aziendale e per gli investimenti
Etica del business, antiriciclaggio e lotta alla corruzione
Impatti ambientali diretti
Educazione finanziaria e sviluppo delle comunità locali
Gestione e sviluppo del capitale umano
Gestione responsabile della catena di fornitura
SDGs a cui Azimut si impegna a contribuire
Legenda
Tematiche ambientali
Tematiche sociali
Tematiche di governance

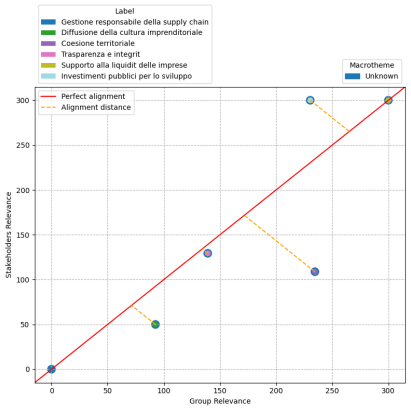
(a)


Label
E sviluppo
E lotta alla corruzione
Stakeholder
E sostenibilit del etica del
Marketing responsabile comunicazione trasparente
Portafoglio prodotti
Impegna a contribuire
Dei dati
A cui azimut si
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

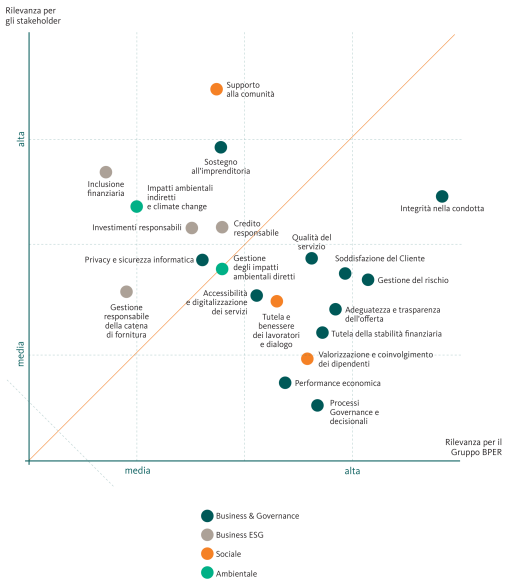
(b)


Rilevanza per Invitalia
Rilevanza per gli stakeholder
Supporto alla liquidità delle imprese
Accelerazione e controllo sull'impiego dei fondi comunitari e nazionali
Benessere delle persone
Rafforzamento del sistema produttivo
Accelerazione investimenti pubblici per lo sviluppo
Rafforzamento dell'azione delle PA
Riqualificazione e valorizzazione dei territori
Creazione e salvaguardia dei posti di lavoro
Sostegno alla coesione territoriale
Utilizzo sostenibile delle risorse energetiche
Trasparenza e integrità
Digitalizzazione
Diffusione della cultura imprenditoriale
Gestione responsabile della supply chain

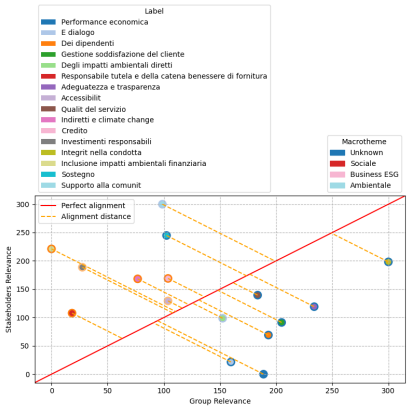
(a)


Label
Gestione responsabile della supply chain
Diffusione della cultura imprenditoriale
Coesione territoriale
Trasparenza e integrit
Supporto alla liquidit delle imprese
Investimenti pubblici per lo sviluppo
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

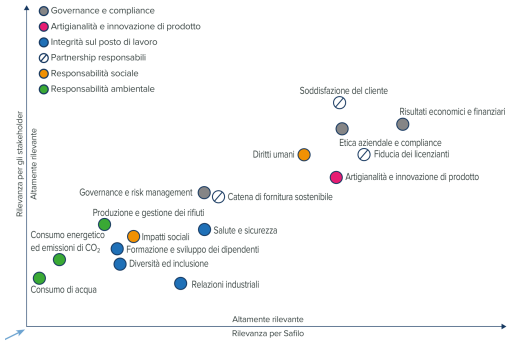
(b)


Rilevanza per gli stakeholder
Supporto alla comunità
Sostegno all'imprenditoria
Inclusione finanziaria
Impatti ambientali indiretti e climate change
Integrità nella condotta
Investimenti responsabili
Credito responsabile
Qualità del servizio
Privacy e sicurezza informatica
Gestione degli impatti ambientali diretti
Soddisfazione del Cliente
Gestione del rischio
Accessibilità e digitalizzazione dei servizi
Gestione responsabile della catena di fornitura
Tutela e benessere dei lavoratori e dialogo
Adeguatezza e trasparenza dell'offerta
Tutela della stabilità finanziaria
Valorizzazione e coinvolgimento dei dipendenti
Performance economica
Processi Governance e decisionali
alta
media
media
alta
Rilevanza per il Gruppo BPER
Business & Governance
Business ESG
Sociale
Ambientale

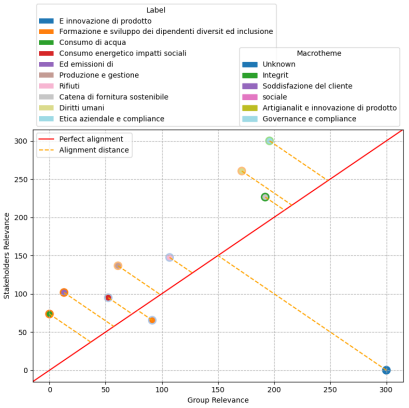
(a)


Label
Performance economica
E dialogo
Dei dipendenti
Gestione soddisfazione del cliente
Degli impatti ambientali diretti
Responsabile tutela e della catena benessere di fornitura
Adeguatezza e trasparenza
Accessibilit
Qualit del servizio
Indiretti e climate change
Credito
Investimenti responsabili
Integrit nella condotta
Inclusione impatti ambientali finanziaria
Sostegno
Supporto alla comunit
Macrotheme
Unknown
Sociale
Business ESG
Ambientale
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

(b)


Governance e compliance
Artigianalità e innovazione di prodotto
Integrità sul posto di lavoro
Partnership responsabili
Responsabilità sociale
Responsabilità ambientale
Soddisfazione del cliente
Risultati economici e finanziari
Etica aziendale e compliance
Fiducia dei licenzianti
Diritti umani
Artigianalità e innovazione di prodotto
Governance e risk management
Catena di fornitura sostenibile
Produzione e gestione dei rifiuti
Salute e sicurezza
Consumo energetico ed emissioni di CO_2
Impatti sociali
Formazione e sviluppo dei dipendenti
Diversità ed inclusione
Relazioni industriali
Consumo di acqua
Rilevanza per gli stakeholder
Altamente rilevante
Altamente rilevante
Rilevanza per Safilo

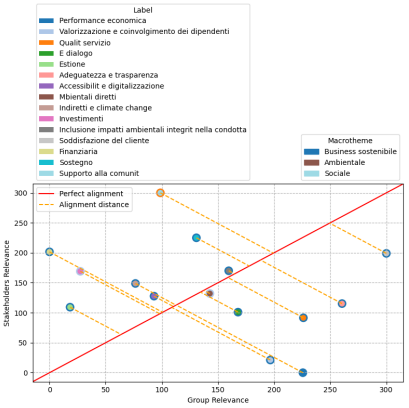
(a)


Label
E innovazione di prodotto
Formazione e sviluppo dei dipendenti diversit ed inclusione
Consumo di acqua
Consumo energetico impatti sociali
Ed emissioni di
Produzione e gestione
Rifiuti
Catena di fornitura sostenibile
Diritti umani
Etica aziendale e compliance
Macrotheme
Unknown
Integrit
Soddisfazione del cliente
sociale
Artigianalit e innovazione di prodotto
Governance e compliance
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

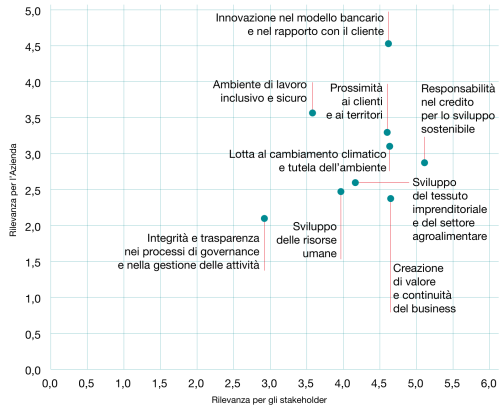
(b)


Rilevanza per gli stakeholder
alta
media
Supporto alla comunità
Sostegno all'imprenditoria
Inclusione finanziaria
Impatti ambientali indiretti e climate change
Credito responsabile
Integrità nella condotta
Investimenti responsabili
Gestione degli impatti ambientali diretti
Qualità del servizio
Soddisfazione del Cliente
Privacy e sicurezza informatica
Gestione del rischio
Accessibilità e digitalizzazione dei servizi
Gestione responsabile della catena di fornitura
Tutela e benessere dei lavoratori e dialogo
Adeguatezza e trasparenza dell'offerta
Tutela della stabilità finanziaria
Valorizzazione e coinvolgimento dei dipendenti
Performance economica
Processi Governance e decisionali
Rilevanza per il Gruppo BPER
media
alta
Business & Governance
Business sostenibile
Sociale
Ambientale

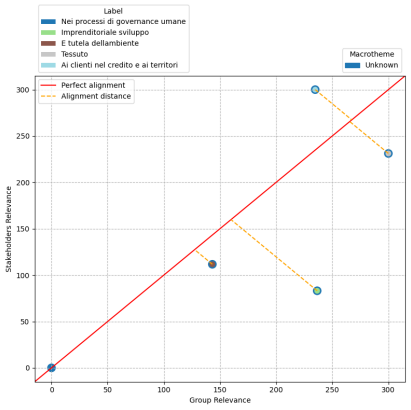
(a)


Label
Performance economica
Valorizzazione e coinvolgimento dei dipendenti
Qualit servizio
E dialogo
Estione
Adeguatezza e trasparenza
Accessibilit e digitalizzazione
Mbientali diretti
Indiretti e climate change
Investimenti
Inclusione impatti ambientali integrit nella condotta
Soddisfazione del cliente
Finanziaria
Sostegno
Supporto alla comunit
Macrotheme
Business sostenibile
Ambientale
Sociale
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

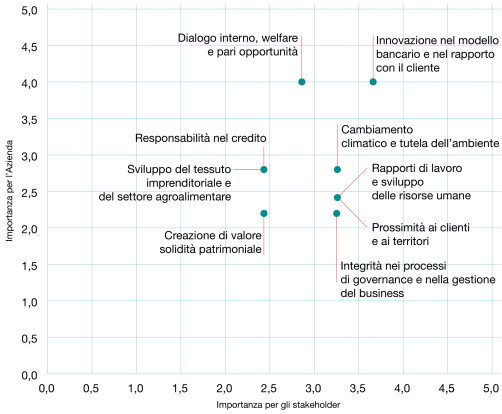
(b)


Innovazione nel modello bancario
e nel rapporto con il cliente
Ambiente di lavoro
inclusivo e sicuro
Prossimità
ai clienti
e ai territori
Responsabilità
nel credito
per lo sviluppo
sostenibile
Lotta al cambiamento climatico
e tutela dell'ambiente
Sviluppo
del tessuto
imprenditoriale
e del settore
agroalimentare
Integrità e trasparenza
nei processi di governance
e nella gestione delle attività
Sviluppo
delle risorse
umane
Creazione
di valore
e continuità
del business
Rilevanza per l'Azienda
Rilevanza per gli stakeholder

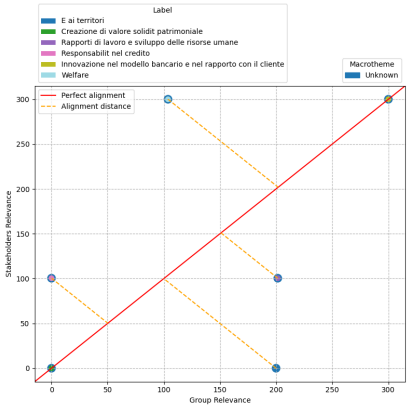
(a)


Label
Nei processi di governance umane
Imprenditoriale sviluppo
E tutela dellambiente
Tessuto
Ai clienti nel credito e ai territori
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

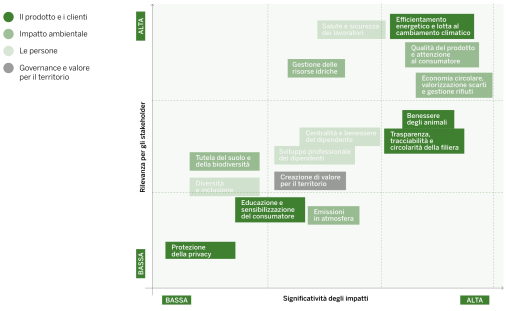
(b)


Dialogo interno, welfare
e pari opportunità
Innovazione nel modello
bancario e nel rapporto
con il cliente
Responsabilità nel credito
Cambiamento
climatico e tutela dell'ambiente
Sviluppo del tessuto
imprenditoriale e
del settore agroalimentare
Rapporti di lavoro
e sviluppo
delle risorse umane
Creazione di valore
solidità patrimoniale
Prossimità ai clienti
e ai territori
Integrità nei processi
di governance e nella gestione
del business
Importanza per l'Azienda
Importanza per gli stakeholder

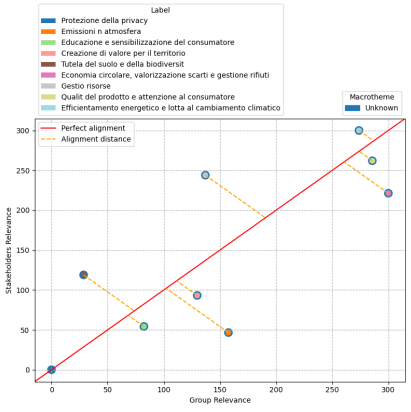
(a)


Label
E ai territori
Creazione di valore solidit patrimoniale
Rapporti di lavoro e sviluppo delle risorse umane
Responsabilit nel credito
Innovazione nel modello bancario e nel rapporto con il cliente
Welfare
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

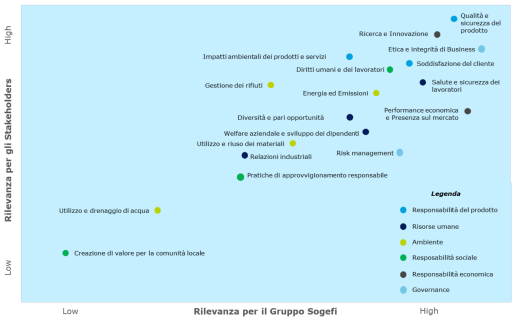
(b)


Il prodotto e i clienti
Impatto ambientale
Le persone
Governance e valore
per il territorio
ALTA
BASSA
Rilevanza per gli stakeholder
Salute e sicurezza
dei lavoratori
Efficientamento
energetico e lotta al
cambiamento climatico
Qualità del prodotto
e attenzione
al consumatore
Gestione delle
risorse idriche
Economia circolare,
valorizzazione scarti
e gestione rifiuti
Benessere
degli animali
Centralità e benessere
dei dipendenti
Trasparenza,
tracciabilità e
circolarità della filiera
Sviluppo professionale
dei dipendenti
Tutela del suolo e
della biodiversità
Diversità
e inclusione
Creazione di valore
per il territorio
Educazione e
sensibilizzazione
del consumatore
Emissioni
in atmosfera
Protezione
della privacy
BASSA
Significatività degli impatti
ALTA

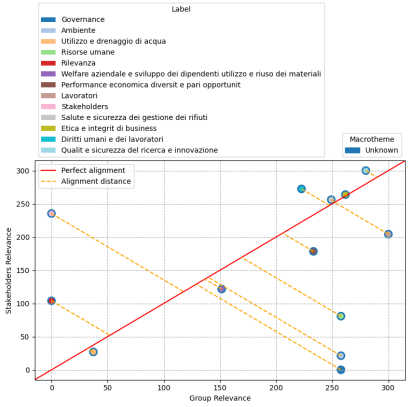
(a)


Label
Protezione della privacy
Emissioni n atmosfera
Educazione e sensibilizzazione del consumatore
Creazione di valore per il territorio
Tutela del suolo e della biodiversit
Economia circolare, valorizzazione scarti e gestione rifiuti
Gestio risorse
Qualit del prodotto e attenzione al consumatore
Efficientamento energetico e lotta al cambiamento climatico
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance

(b)


High
Rilevanza per gli Stakeholders
Low
Qualità e sicurezza del prodotto
Ricerca e Innovazione
Etica e integrità di Business
Impatti ambientali dei prodotti e servizi
Soddisfazione del cliente
Diritti umani e dei lavoratori
Salute e sicurezza dei lavoratori
Gestione dei rifiuti
Energia ed Emissioni
Performance economica e Presenza sul mercato
Diversità e pari opportunità
Welfare aziendale e sviluppo dei dipendenti
Utilizzo e riuso dei materiali
Risk management
Relazioni industriali
Pratiche di approvvigionamento responsabile
Legenda
Utilizzo e drenaggio di acqua
Responsabilità del prodotto
Risorse umane
Creazione di valore per la comunità locale
Ambiente
Resposabilità sociale
Responsabilità economica
Governance
Low
Rilevanza per il Gruppo Sogefi
High

(a)


Label
Governance
Ambiente
Utilizzo e drenaggio di acqua
Risorse umane
Rilevanza
Welfare aziendale e sviluppo dei dipendenti utilizzo e riuso dei materiali
Performance economica diversit e pari opportunit
Lavoratori
Stakeholders
Salute e sicurezza dei gestione dei rifiuti
Etica e integrit di business
Diritti umani e dei lavoratori
Qualit e sicurezza del ricerca e innovazione
Macrotheme
Unknown
Perfect alignment
Alignment distance
Stakeholders Relevance
Group Relevance
0
50
100
150
200
250
300

(b)

4.3 Final considerations

Let's consider the results showed in the previous sections.
As can be seen in **Table 4.3**, the application has correctly extracted the majority of the images. However, as these statistics are automatically defined by the program by using a computer vision-based heuristic (the one mentioned in sections **(3.1.2.4)** and **(3.1.2.5)**), it may happen that the image is detected as "correctly extracted" while actually being ambiguous (this is caused by the high variability of the reports' style).
On the other hand, as shown in **Table 4.4**, the table extraction performs very well, being able to recognize GRI-type tables by just looking at each page's header.

Regarding the second part of the process involving the materiality matrices interpretation – as highlighted by the images portrayed in section **(4.2)** – the majority of the materiality matrices are correctly converted, with only some of them (like the last one shown in the previous section) having some interpretation problems because of their particular shape. Moreover, some of the labels seem to be lacking of words, depending from the resolution of the input matrix or from the accuracy of the chosen OCR routine; nevertheless, the acquired numeric data seem overall to agree the trend shown in the sources matrices and could easily be contextualized.

Summarizing, this solution may certainly be helpful during the visual data extraction, but it should not be seen as a complete replacement of a human operator, which could use this solution as a "first trace" of a supervised process. With further development and refinement, this solution could certainly be useful in corporate sustainability analysis contexts.

Bibliography

[DH72] Richard O. Duda and Peter E. Hart. "Use of the Hough Transformation to Detect Lines and Curves in Pictures". In: *Commun. ACM* 15.1 (Jan. 1972), pp. 11–15. ISSN: 0001-0782. DOI: 10.1145/361237.361242. URL: https://doi.org/10.1145/361237.361242.

[Hou62] Paul VC Hough. *Method and means for recognizing complex patterns*. US Patent 3,069,654. Dec. 1962.

[IBM] IBM. *What is natural language processing?* Last access on Oct 15th 2023. URL: https://www.ibm.com/topics/natural-language-processing.

[Inn23] Redazione Osservatori Digital Innovation. *Natural Language Processing (NLP): come funziona l'elaborazione del linguaggio naturale*. Last access on Oct 15th 2023. Apr. 2023. URL: https://blog.osservatori.net/it_it/natural-language-processing-nlp-come-funziona-lelaborazione-del-linguaggio-naturale.

[Lin93] Tony Lindeberg. "Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention". In: *International Journal of Computer Vision* 11.3 (1993). QC 20130423, pp. 283–318. DOI: 10.1007/BF01469346. URL: http://link.springer.com/article/10.1007%2FBF01469346.

[Smi07] R. Smith. "An Overview of the Tesseract OCR Engine". In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. Vol. 2. 2007, pp. 629–633. DOI: 10.1109/ICDAR.2007.4376991.

[Tho21] A. Thomas. *Optical character recognition*. Last access on Oct 15th 2023. Feb. 2021. URL: https://medium.com/sfu-cspmp/optical-character-recognition-948bfc4adfb3.

Bibliography

[Wik23] Wikipedia. *Levenshein distance*. Last access on Oct 15th 2023. Sept. 2023. URL: https://en.wikipedia.org/wiki/Levenshtein_distance.

[Zan22] L. Zanotti. *Matrice di materialità: Cos'è e perché l'analisi è importante*. Last access on Oct 15th 2023. Dec. 2022. URL: https://www.esg360.it/esg-world/matrice-di-materialita-cose-come-si-fa-importanza-analisi/.