EasyChair Preprint

№ 934

An adaptive agent for Google Place crawling

Domenico Monaco

May 26, 2019

An adaptive agent for Google Places crawling

Domenico Monaco[1]

monaco.d@gmail.com

Abstract

Intelligent agents are used in different academic and professional areas for various scope, one of this is Marketing and Social Media. With this work is described an adaptive agent for Google Places crawling based on Earth space honeycomb tessellation [6] developed to resolve the first question of my Master Thesis Master [13]: *crawling all Google Place of an urban area in order to feeds geospatial marketing analysis.*

In the context of this work the urban area is the real environment while the Google Place API is a digital representation of it where the agent goal is capturing all places of an area with a minimum input. [4]

The agent, in completely autonomy and with a minimum input, capture all places starting from a central point, by means of a spiral movement up to a maximum diameter, both specified by user. This spiral movement (spiral-pattern) [8] it is made over an honeycomb tessellation [7].

The agent behaviours are characterized by: planning of movement path, collecting and storage of places, checking of results, if necessary adapting of it behaviours, fault tolerance and replanning of actions. All of this by the minimum user-input are composed by: center of crawling, default size of cells and finally the number of spirals of crawling.

The core of algorithm is the adaptation on some environmental details of his planned track and granularity of tessellation previously planned. The algorithm choose when use more smaller cells and where, and, if there are some problem, where re-planning cells.

Keywords: intelligent agent, data scraping, geospatial, social media, urban analysis, honeycomb tessellation

Contents

List of Figures

1 Introduction

The term Social Media is changed during the last years, but in general can be defined as "as web-based and mobile-based Internet applications that allow the creation, access and exchange of user-generated content that is ubiquitously accessible". [2]

More precisely, and historical definition, Social media is defined as a group of internet-based applications that build on the ideological and technical foundations of Web 2.0, and that allow the creation and exchange of user generated content. [9]

Today we have different examples of social media platforms that allow people to interact and share content usually for specific purpose as video sharing, photo sharing, geolocated information, work information or generic purpose.

1.1 Marketing: from Big Data to Smart Agent

Current Social Medias are become more a more pervasive in every aspect of our life, just think to more and more social networks that are vital in our daily activity.

In this way the complexity of real world is progressive replicated in digital environments composed by large complex dataset of structured and unstructured information that today we know as "Big Data".

These Big Data, today, is one of most investigated research area because their growed use thanks the numerous AI algorithm that promise to use this data in new ways to improve several business and accadamics goals.

Most of business areas are based on decision making strongly relate to data such as Marketing and Business Intelligence that today thanks to Social Media, the consequent Big Data and finally thanks to AIs techniques can improves the quality of their decision. [11]

But all of this, to work correctly, must to have a lot of datas, possibly of good quality.

Smart Agent and Marketing By Marketing point of view, in most cases, these huge quantity of data are to feed smart agents that support, help or cooperate with professional or with other smart agents during the complex flow of Business process. [10]

How to Kumar et al. explain, there are several type of agent that are used in marketing, in particular these can be divide in: Info Acquistion , Analysis, Decision making, interaction/negotiation, Collaboration. That are used for specific marketing purpose area as: Market Governance, Customer orientation, Competitor orientation, Learning orientation.

How we can see, at the base of this agents there is the "information acquisition" that enables several other agents or business process to working. One of this information acquisition can be executed through the Social media scraping.

Volunteered geographic information

Volunteered geographic information (VGI) is *a special case of the more general Web phenomenon of user-generated content* [3], today not only related to WebGis but more generally present into a lot of Social Media. This enable to share geographic information via the Internet that create user-generate geographic knowledge [3]. Some examples of such social media specifically designed for geographic information are: Google Map[1], Bing Map[2], Facebook Places[3],

[1] http://maps.google.com/
[2] https://www.bing.com/maps
[3] http://facebook.com/

Foursquare [4] and TripAdvisor[5].

All the geographic information made available by Social Media is a union between VGI, administrative information, generated by algorithms or edited by the company that provides the Social Media service

Place The central geographical element of a Social Media is the so-called Place (POI), the term translates into a place, but it can also be a monument, a building, a park or even an area of the city; therefore it can be punctual or more extensive. In other words, it is anything on the earth's surface that can be identified as a coordinate or boundary.

These places are equipped with a large amount of information such as photos, descriptions, opening and closing times, telephone numbers and user reviews.

Given all these characteristics, the Place acquires a strategic importance with a view to carrying out geospatial marketing analysis.

1.2 Social Media Scraping

How Batrinca and Treleaven expresses in his research: "Social media data is clearly the largest, richest and most dynamic evidence base of human behavior, bringing new opportunities to understand individuals, groups and society [...] and the scientists and industry professional are increasingly research novel ways of automatically collecting of data.

One of technique to gather data is "Social media Scraping". Historically, Scraping or Collecting is typically part of Social Media analysis, but today is also important for several area based on user-data such as Machine Learning, Deep learning and AIs in general.

But cause of increasingly attention of data this practice has characterized by new challenges that scientists and professional must resolve.

In particular although social media data is free accessible through APIs, due to the commercial value of the data and privacy, most of the major social Media are making it increasingly hard for academics to obtain complete access to their data by introducing restricted limitation and rules. [2]

In addition I add that: the increasing of complexity, velocity of changing and quantity of Social Media datas introduce new complexities into scraping of datas as example the difficulty of one-shot scraping, prediction of quantity of data, duration of scraping process.

One another interesting aspect of Social Media Scraping is the "transparency of scraping process", because the quality the methodology of scraping can be critical to the social research that use the gathered data. As expresses Marres and Weltevrede: "A significant amount of online social research is based on non-disclosed data sets, and it has been argued that this opacity should be explicitly challenged: good digital social research should open source the code, or at least provide pseudo code explaining the full recipe of extracting, cleaning and ordering the data." [12]

Social media scraping as agent problem

By starting these considerations, the idea that at the basis of this work is developing a Social Media crawling data process, considering the Social Medias as complex environments in a same way of real world and consequent approach of crawling as intelligent agent that adapt it behaviours in base of context of crawl and in according to predefined goals of crawling.

[4] https://it.foursquare.com/
[5] https://www.tripadvisor.it/

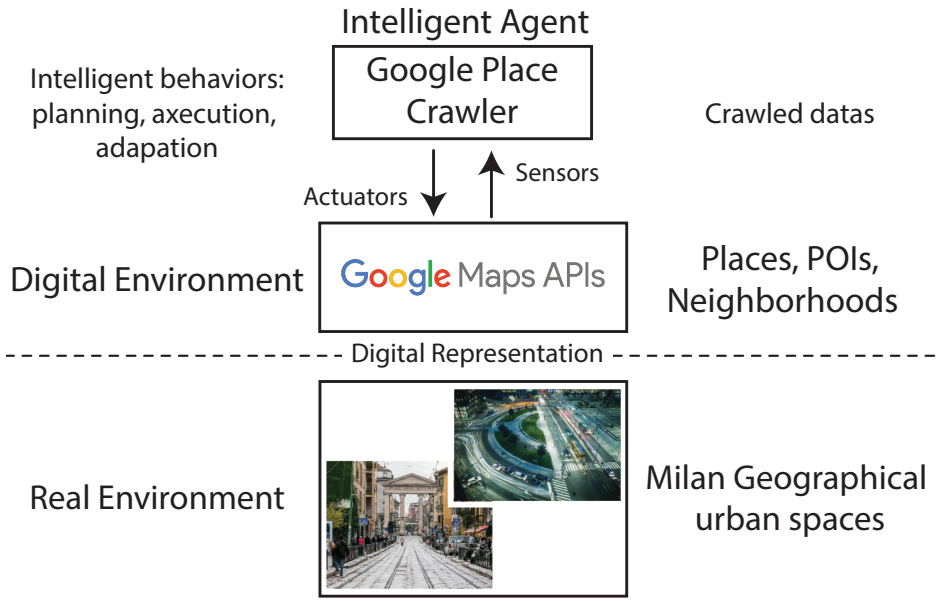

Figure 1: Google Place scraping as intelligent agent

In particular, an simple crawler that has a goal to acquire all place of specified area by Google Place API, must became and intelligent crawler that work as an agent in real world, with a set of Actions, Behaviours, Adaptation and Fault tolerance skills; all of this with minimum user-input.

2 Environment

In this work we have two level of environments: the real world and his partial representation, the digital environment (see figure 1).

Digital and Real Environment The real environment in this case is a urban area with many areas, places, points of interest and roads classically represented over a Geospatial Social Networks like a Google Map, OpenStreetMap and more. These are the our digital Environments, as digital representation of real world. They have some rule of access and limitations, sometimes similar to real world, sometime very different.

In our case the digital environment is Google Place API, that enable people to accessing to places, areas and roads located on Earth surface by Web API.

2.1 Rule of access and limitations

In particular, to accessing on Google Map platform, the Google Place API [4] provides the following types of requests:

- Place Search: that returns a list of places based on a user's location or search string;

- Place Details: that returns more detailed information about a specific place, including user reviews;
- Place Photos: that provides access to the millions of place-related photos stored in Google's Place database;
- Place Autocomplete and Query Autocomplete: that provides specifics autocomplete function for search bars;

In according to our main goal, the places gathering of an urban area, we take attention primarily on "Place Search" request type. This represent for us the main mode of accessing on the environment, so Sensors and Actuators of agent must be work over this method of access.

Place Search

The Places Search API [5] allows users to search for places either by proximity (named Nearby Search) or a text string, both returns a list of places along with summary information about each place.

Nearby Search The API URL of Nearby Search mode is:

`https://maps.googleapis.com/maps/api/place/nearbysearch/output?parameters`

The Output can be XML or JSON, while parameters can are required or optional.

Required Parameters Required parameters consist in a set of values that characterize of our access on digital environment. [5]

- key - is user identification key by which depends the levels of access, maximum number hourly request and other limitations;
- radius - is the distance (in meters) within which to return place results, the maximum allowed radius is 50000 meters; this characterize the access on environment as an "circle view";
- location - is the center request around which to retrieve place information, is specified as latitude, longitude.

Limitations of accessing By default, each Nearby Search or Text Search returns up to 20 establishment results per query; however, each search can return as many as 60 results, split across three pages, these accessible page-by-page by *next_page_token* value in relative previous page. [5]

2.2 How the crawler sees the Google Place digital environment

In our case, with Google Place API Nearby Search the crawler can access to digital environment as partial representation of real environment through particular view of geographical area (see figure 2).

This access must respect some rules and limitations, in particular every single access are characterize by:

- is a circle view on geographical area, with latitude/longitude centers;

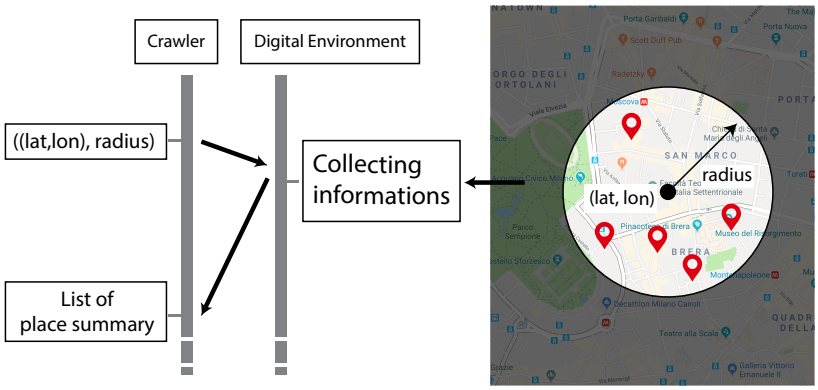

Figure 2: Example of access on digital environment through Google Place API

- this circle can have maximum 50000meters of radius;
- can contain maximum 60 places divided in 3 pages;
- that depends by our buyed access (es example free or business), we have hourly limitations on number of accesses.

Unpredictables changes and intelligent behaviors Our digital environment can have unpredictable changes or results at the time of crawling start such as: the density of places in specific area, therefore the numbers of pages or the places losing. Other unpredictables change can be is the error during the request, Google blocking action in according to the our access levels or size of different area interested by crwaling. All of these can amplify or decrease the unpredicatible changes.

These characteristics create the environment rules where our crawler, in according to goals, must work independently by the area, position, results or faults. In this way the crawler must is an intelligent agent that adapt their behaviours by the unpredictables changes of context.

3 Adaptive Agent

In this work i name the crawler adaptive, not intelligent, because the intelligence is too strong concept to be used in this case. It's sure adaptive. It's intelligent? Maybe after future works, because i see that the "adaptation" a component of "intelligence".

Agent goal Our adaptive agent have as goal acquiring all of places located in a such area of Earth (in this case Milan city) in respect of previous environmental rules and limitations of environment access, with a minimum user-input and in completely autonomy

3.1 Agent Environment representation

The agent to work correctly must to have a representation of environment where it work, in this case is *secondary* digital representation derived directly by Google Place API. This internal representation of environment can be seen as another level of real environment, in this case

derived by its digital representation (see figure 1). In all of this stack of representations some complexity are added and other are deleted.

In this case, according to environment rules, limitations and agent goal the agent representation is *is a discrete tessellation that is able to cover the space of an geographical area taking into account the circular shape of accessing method.*

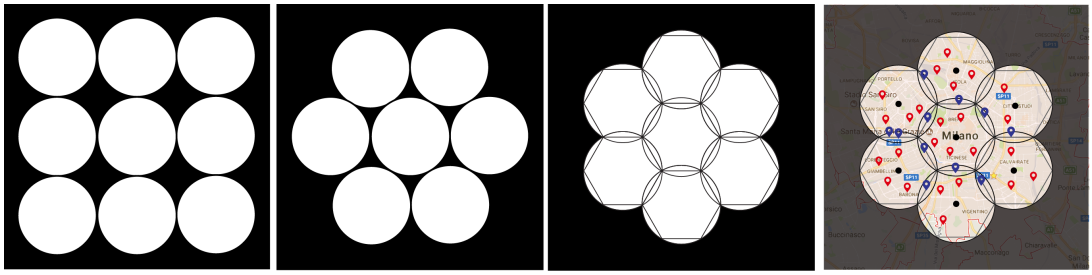

Figure 3: Hexagons tessellation as approximation of circle packing

From circle packing to hexagons tessellations The agent see the environment through an circle-view, so instinctively we can solve the problem of "representation" as an "circle packing". But in this case is not a viable solution thanks to the complexity of generation of tessellation based on an well designed "circle packing". Is too complex to is manged by our agent.

Therefore I've decided to use an hexagon tessellation (honeycomb tessellation), where each hexagon are an approximation of circle. With this solution is possible to solve the "circle packing" problem with simple tessellation. This, at the expense to a greater (and unavoidable) overlapping of "agent views."

The agent environment representation as hexagon tessellation, in according to [7], is the best approximation of circle with a minimum overlapping between circles (see figure 3). In addition, respect to others shapes tessellation, we can have same distance between neighboring hexagons.

3.2 Actions

As an human moves in the world step-by-step through walking action, our agent can moves in its environment hexagon-by-hexagon at fixed size through its basic actions.

The agent must basically be moving over his discrete representation, therefore the base actions and environment representation generation can be fused in six base movement action based on initial position.

Considering that a general position of the agent in the space, it must:

- have perception of accessible new positions;
- decide where it will moves in base the previous perception.

The basic six movement possible in a hexagon tessellation from a central hexagons are: north (NN), northeast (NE), east (EE), southeast (SE), south, southwest (SW), west (WW), northwest (NW). All other movements are directly derived from these as composition or repetitions of these (see 4).

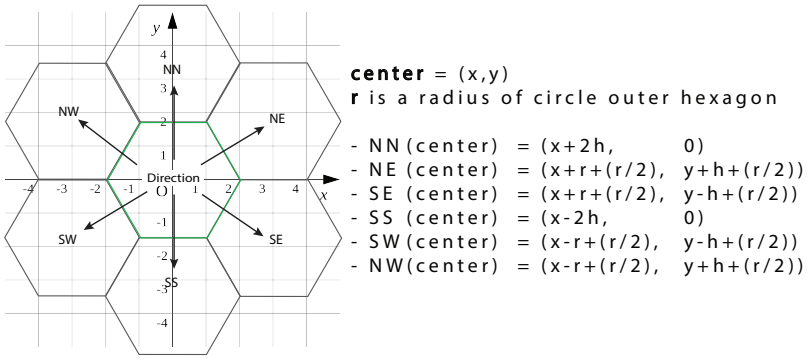

Figure 4: Basic six movements of agent over hexagon tessellation

These are directly derived by faces of central hexagon therefore the points of contact of near hexagons. This exemplification of action is clearly impossible to have if we had used the circle packing.

Great-circle Distance The "Great-circle Distance" is the way to measure the distance between two geographical points on the Earth, it consists in calculating the length of the arc of the largest circumference that passes between these two points, this circumference is called the maximum circle distance. It is also called spherical distance, it is the shortest distance between two points on the Earth calculated on the reference sphere of average radius 6,371 km (see figure 5).

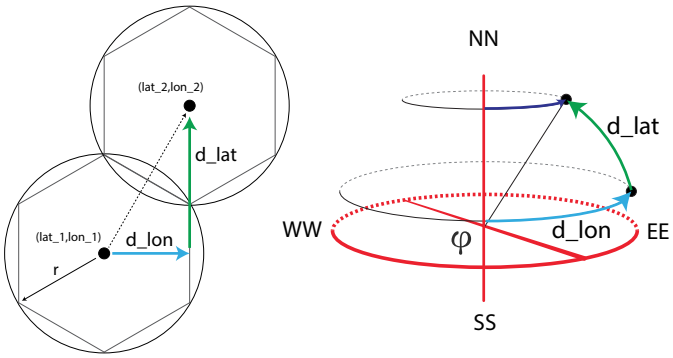

Figure 5: Exemplification of movement of crawler calculated with adaptation of meters inLatitude and Longitude degree by using Great-Circle Distance

In a first moment the agent predict the meters of movement over horizontally and vertically coordinate. In a second moment transform this meters values in longitude an latitude degree movements thanks to Haversine Formula.

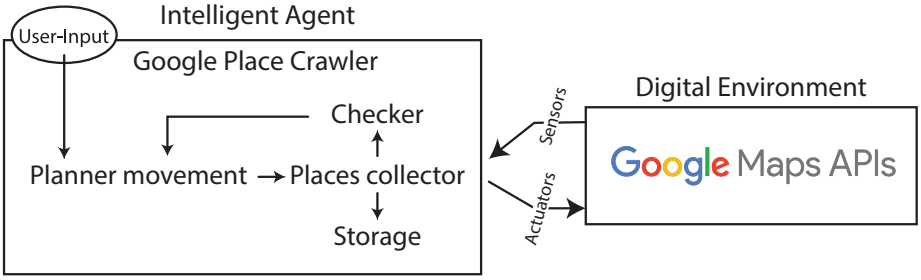

Figure 6: Model of Google Place Crawler as intelligent agent and its behaviours

3.3 Behaviour

Like a intelligent entity the our agent must have some type of behaviour. According to Allen et al. [1] the human movements are managed by brain through phases: planning, execution and perception of results of action. In more simple way we can consider a movement composed by a phases of planning and execution.

In a same way the our agent plan the future actions, execute one action per-time and verify the output (see 6). At this point, in according to the environmental feedback, the agent decide if execute the next planned action or add to the personal "stack of planned action" a new set of actions.

Spiral-pattern movement behaviour

In according to environmental rules, limitation and especially for the his goal the basic behaviour is a spiral-pattern movement inspired by Roomba initial exploration movement [8].

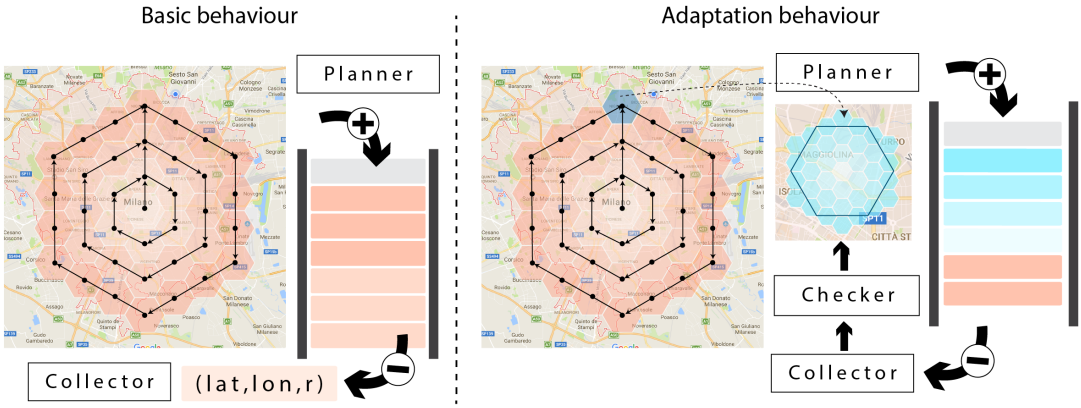

Figure 7: Basic behaviour and adaptation behaviour

Because, in this case, the agent environment representation is an hexagon tessellation, the basic behaviour is implemented with a FIFO Queue that is filled with hexagon center position. From the first hexagon in initial position, and progressively with the other hexagon following a spiral-pattern (see figure 7).

In other words the agent, starting from the center, calculate the position of every hexagons ring-by-ring. Where the initial hexagon for the i-th ring is the one positioned in top of ring.

Adaptation

The adaptations of agent is directly derived from the limitation of 60 maximum Place for request. Because, we can't know the density of every area in according to the previous limitation.

For this reason the agent, in a case of detection of 60 places in a single hexagon, he plan a new small spiral with more fine grained hexagon that overlap the hexagon where are detected a 60 Place limitation (see figure 7).

Fault tolerance

The fault tolerance is necessary to achieve the autonomy of agent, in particular its fault tolerance is initially developed to avoid typically problems of technology used by Google Place API.

In fact when we using the Web API is possible that we have some problem such as communication errors, wrongs API parameters and some other problem derived of internet protocols communications or API access policy.

In this case the agent actuate a conduct aimed at correcting the error with re-planning the failed action, in this case failed collecting of places in some specific hexagons.

In this work an failed collection can be caused by: exhausted or banned Google user-key or communication errors.

Stop and Restart abilities To avoid loss of work the agent is able to stop and restart their crawling without repeating acquisitions. In particular is possible to: programming crawling by defining initial and final ring of crawling to achieve step-by-step crawling (see figure 8), restart the crawling by last area collected after an hard reset of agent or machine where it is active.

3.4 User-input

To work the agent must be have some basic inputs, in particular is defined an user-input that instruct the agent especially in entry point of actions (see figure 6), this must is a minimum possible to don't asking too a lot information to the user.

The user-input of agent are composed by:

- initial center position of crawling, in this case in GPS coordinate;
- the radius of default circle that is approximated to the hexagon, in this case in meters;
- the numbers of ring of spiral-pattern;
- division ratio of fine grained hexagons tessellation for the adaptation behaviour, that for default is 1/3 of default radius circle

By these inputs the adaptive agent can crawl the totally Places over any area of Earth, in this case using a Google Place API, without losing place in a some area with high density or caused by error.

4 Conclusion and future works

4.1 Results

In conclusion, the agent was tested for more than 6 month on Google Place API over 52 Km of diameter from Milan to near city like Monza and others. It has captured without losses over 290000 Places applying adaptation and fault tolerance behaviours in completely autonomy.

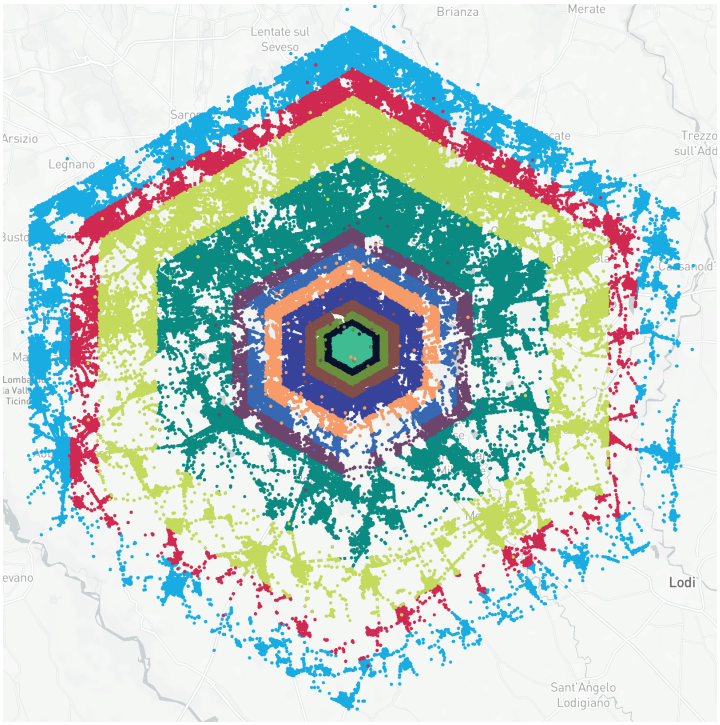

Figure 8: The over 290000 Places, from Milan area to near cities, divided in different step of crawling;

For simplicity and first using in debug mode, the agent was equipped with a possibility of execute different portion of a spiral movement, as example in a first time from ring 10 to 20, and in a second moment from 21 to 40. (figure 8)

The adaptation behaviour in action The adaptation can't be "predicted", depends by the local densities of places, in according to the algorithm default request radius.

As we can see in figure 9 there is a basic tessellation of crawling that cover regularly the space with some areas with more fine grained tessellation. These, as we can see in example, are composed by some added centers with less distance between. Where apply more fine crawling path is decided in real time during the execution of crawling, in autonomously mode, by the evaluation of Environment conditions that in this case represented by the number of crawled places.

The default distance between crawling centers, in this specific example, is 60 meters while the distance between the more fine grained crawling centers is 20 meters (1/3 of default size).

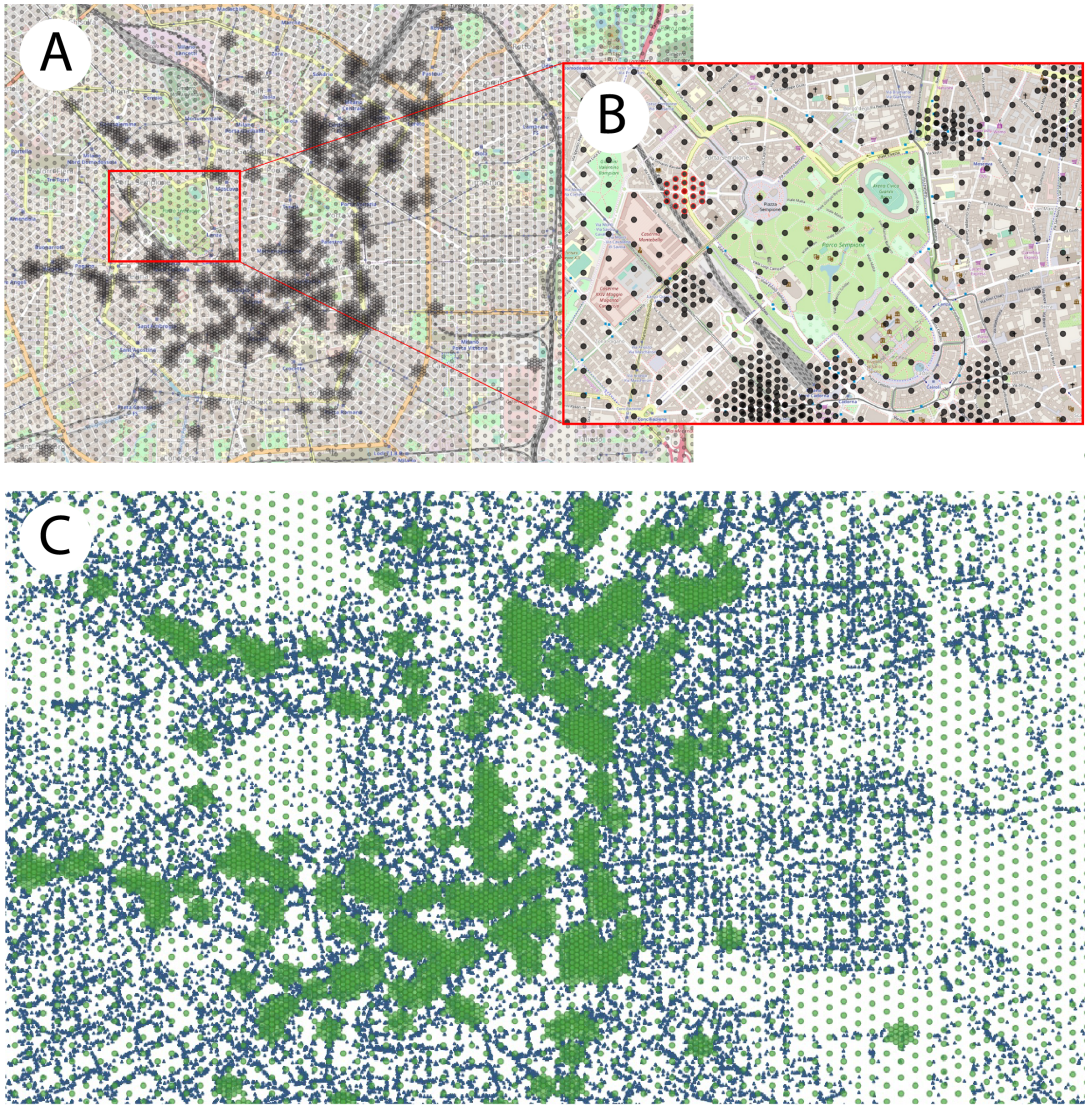

Figure 9: (A) Centres of crawling of a Milan area portion with particular view (B) of Adapted granularity of crawling (points in red); (C) Comparison of crawled POIs (blue points) and centers of crawling (green points)

The default size was decided by empirical tests that showed as 60 meters is best solution to do not have too high number of sub-tessellation and in a same time to do not have too high number of with empty requests.

4.2 Different uses

Clearly, other cities where is possible to collect Points of Interest is the primary different use, this is partially in action and in program to be actuated in more systematic way. But in more

strong way is possible to think very different uses of the basic idea.

Different behaviours

We can imagine other behaviors base to the six degree basic movement as a custom path pattern that follow a different goal of this paper. This because the hexagon tessellation is very powerful representation of a plane space with the equals movement in six different direction with a same weight in a term of distance.

Other applications

Obviously, Google Place API is one of more specific case of digital environments representation. By starting this we can imagine a other application of this agent as different Geolocated Social Network with same (or similar) rules of access.

Or in other case we can generalize the approach to different use case, in example for drones for aerial photo mapping o simply sequential photo capturing over very large spaces where zoom or other factor can be varied during acquisition.

Honeycomb walking pedestrian behavior model

One interesting use is into pedestrian simulations, in particular is possible to derive an Honeycomb walking pedestrian behavior model.

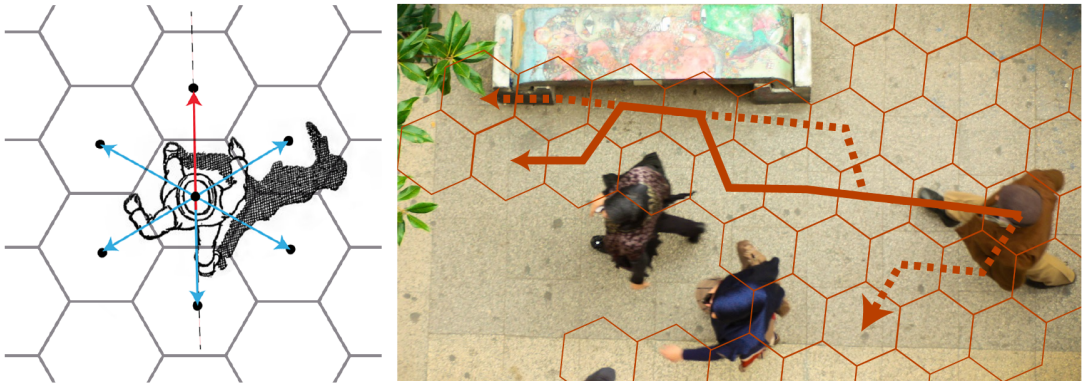

Figure 10: Example of Honeycomb walking pedestrian behavior model

All agent behaviours descripted in this work can increase the quality of movement of an pedestrian agent: as from the equal distance between cells to the varying of cells tessellation (see figure 10).

As example the adaptation behaviours can be used when the agent meets an obstacle and must decide how skip it with more detailed movements, like when an human take attention when crossing a puddle.

The same idea can be used obviously in other context such as autonomous cars that can approximate the planned paths to honeycomb tessellation, where in some case must plan more detailed paths ad in other less fine grained path in order to avoid the calculation bottlenecks when it is not necessary.

Unfortunately this idea was impossible to develop due to lack of time and resource.

4.3 Comparison with spatial data clustering algorithms

In Master Thesis [13] these POIs was used as input for Recursive Data Clustering Algorithm [14] specifically developed to work with geolocated points.

One question that remain open in Master Thesis is:

If exists, what is the relation between areas where the agent has actuated a fine-grained crawling and clusters detected by the Recursive Data Clustering? **(see figure 9).**

The situation that I expect is that fine-grained crawling areas are located near o inside high density cluster detected by my "Recursive data clustering through finding vague solutions" [14].

The second question, derived by the last one is:

If what i expect is verified, is possible think to use this approach to make a Grid-based spatial data clustering ?

References

[1] GI Allen, GB Azzena, and T Ohno. Cerebellar purkyně cell responses to inputs from sensorimotor cortex. *Experimental brain research*, 20(3):239–254, 1974.

[2] Bogdan Batrinca and Philip C Treleaven. Social media analytics: a survey of techniques, tools and platforms. *Ai & Society*, 30(1):89–116, 2015.

[3] Michael F Goodchild. Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69(4):211–221, 2007.

[4] LLC Google. Google places api — google developers, 2017. URL https://developers.google.com/places/. (2017).

[5] LLC Google. Search place google places api — google developers, 2017. URL https://developers.google.com/places/web-service/search. (2017).

[6] Zahia Guessoum. Adaptive agents and multiagent systems. *IEEE Distributed Systems Online*, 5(7), 2004.

[7] Thomas C Hales. The honeycomb conjecture. *Discrete & Computational Geometry*, 25(1): 1–22, 2001.

[8] Joseph L Jones and Philip R Mass. Method and system for multi-mode coverage for an autonomous robot, June 11 2013. US Patent 8,463,438.

[9] Andreas M Kaplan and Michael Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53(1):59–68, 2010.

[10] V Kumar, Ashutosh Dixit, Rajshekar Raj G Javalgi, and Mayukh Dass. Research framework, strategies, and applications of intelligent agent technologies (iats) in marketing. *Journal of the Academy of Marketing Science*, 44(1):24–45, 2016.

[11] Andreina Mandelli. *Big Data Marketing: Creare valore nella platform economy con dati, intelligenza artificiale eIOT*. EGEA spa, 2017.

[12] Noortje Marres and Esther Weltevrede. Scraping the social? issues in live social research. *Journal of cultural economy*, 6(3):313–335, 2013.

[13] Domenico MONACO. Analisi di dati geospaziali per applicazioni di urban informatics: il caso dei google place nella citt di milano. Master's thesis, Universit degli Studi Milano-BICOCCA, 10 2018.

[14] Domenico MONACO. Recursive data clustering through finding vague solutions. Master's thesis, Universit degli Studi Milano-BICOCCA, 10 2018.