A STATISTICAL MODEL TO PREDICT CO_2 SEQUESTRATING CARBONIC ANHYDRASE FOR DIFFERENT INDUSTRIAL PROCESSES

Project Report submitted in fulfillment of major project of

BACHELORS OF TECHNOLOGY IN BIOTECHNOLOGY

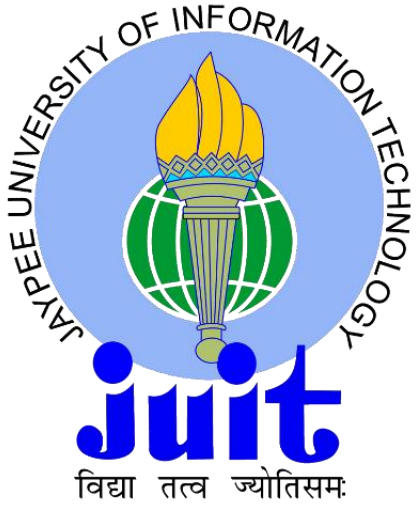
by

NEHA VERMA (171825) and

UMANG MAHESHWARI (171813)

Under the Supervision of

DR. ASHOK NADDA



DEPARTMENT OF BIOTECHNOLOGY AND BIOINFORMATICS

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY WAKNAGHAT,

SOLAN (H.P)

TABLE OF CONTENTS

DECLARATION

We hereby declare that the major project work entitled *"A Statistical Model to Predict CO2 Sequestrating Carbonic Anhydrase for Different Industrial Processes"* has been solely submitted to the Department of Biotechnology and Bioinformatics, Jaypee University of Information Technology, Waknaghat in due of the literature review and research work we have done under the major project in guidance of our supervisor **DR. ASHOK NADDA.**

neha verma

Neha Verma (171825)

Umang Maheshwari

Umang Maheshwari (171813)

Department of Biotechnology and Bioinformatics

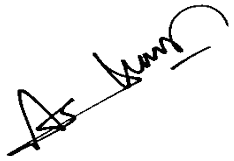
Jaypee University of Information Technology

Waknaghat, Distt-Solan, H.P. – 173 234

Date: 19th May 2021

SUPERVISOR'S CERTIFICATE

This is to certify that the major project work titled *"A Statistical Model to Predict CO2 Sequestrating Carbonic Anhydrase for Different Industrial Processes"* submitted by **Neha Verma and Umang Maheshwari** during their 8th semester in June 2020 in fulfilment for the major project in Biotechnology of Jaypee University of Information Technology, Solan has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of any degree or appreciation.



(Signature of Supervisor)

Name of Supervisor **-Dr. Ashok Nadda**

Designation **-Associate Professor**

Department of Biotechnology and Bioinformatics

Jaypee University of Information Technology

Waknaghat, Distt-Solan, H.P. – 173 234

E-mail -ashok.nadda09@gmail.com

Date **- 20th May 2021**

ACKNOWLEDGEMENT

We take this opportunity to express our first and foremost gratitude to our "DEPARTMENT OF BIOTECHNOLOGY AND BIOINFORMATICS" for the confidence bestowed upon us and entrusting our project title *"A Statistical Model to Predict CO2 Sequestrating Carbonic Anhydrase for Different Industrial Processes"*.

At this juncture, with proud privilege and profound sense of gratitude we feel honored in expressing our deepest appreciation to **Dr. Ashok Nadda,** for being a lot more than just a supervisor and going beyond the call of duty in our guidance, support, advice, and motivation throughout. He has been the source of inspiration of come what may, these issues cannot bring you down. Sincere thanks for his insightful advice, motivating suggestions, invaluable guidance, help and support in successful completion of this major project and also for his constant encouragement and advice throughout our minor project work.

Special thanks to our parents for their infinite patience and understanding and project partners for the constant support and most importantly God, who in his mysterious ways, always made things work out in the end.

In gratitude,

Neha Verma (171825)

Umang Maheshwari (171813)

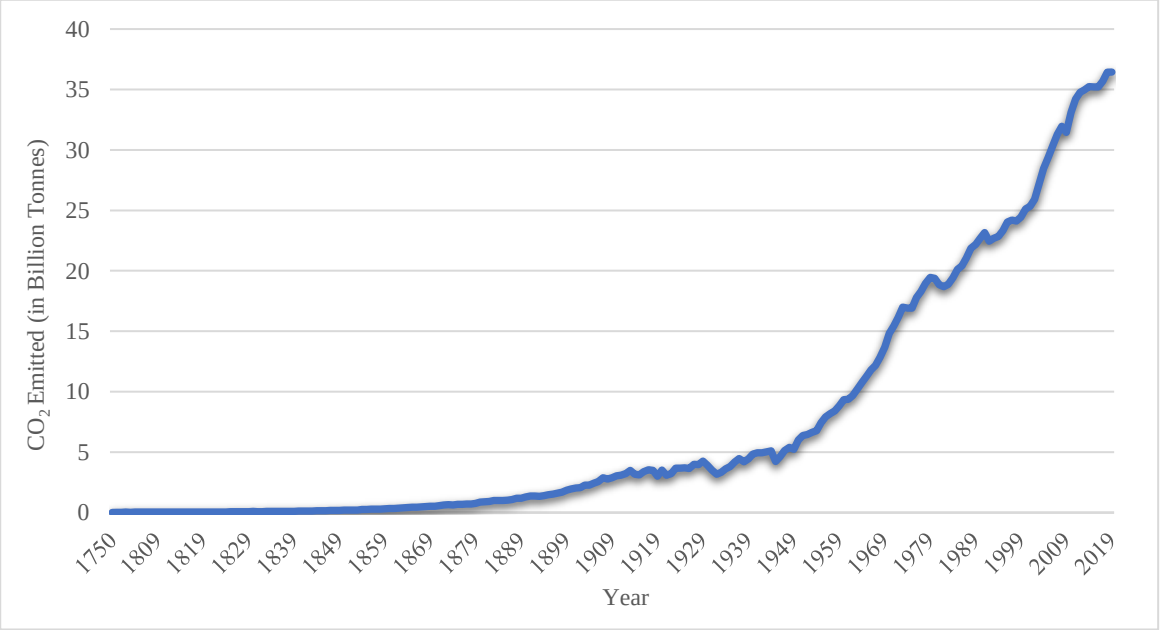
ABSTRACT

Every industrial process requires energy and raw materials, which results in the release of by-products such as liquid effluents, emission of CO_2 or other greenhouse gases. These effluents and emissions have a very adverse impact on the environment. So, the technologies that can eliminate CO_2 or slower its production, will have a huge impact on the improvement of environment, hence will lead to tremendous decrease in greenhouse gases. Carbonic anhydrase is one such metalloenzyme that possess metal in its active center that catalyze the hydration of CO_2 and dehydration of H_2CO_3.This study aims to develop a statistical model that screens the carbonic anhydrase to be used in different industrial processes based on their temperature requirement. The K_{cat} /K_m ratio was used, for selecting a CA producing organism having higher efficiency for conversion of substrate into products. In present study. *Sulfurihydrogenibium azorense,* a Bacteria was chosen as a model organism with highest catalytic efficiency of 350000 ($1/mMs^{-1}$). Similarly, several organisms were selected which grow under varied physicochemical conditions such as temperature, pH and substrate concentration. Using statistics and, graphical analysis this data was plotted and a model was developed that compares and recommends the carbonic anhydrase that can be utilized in various industrial processes based on their optimum temperature.

Keywords: *Carbonic anhydrase, CO_2 sequestration, environmental biotechnology, catalytic efficiency, temperature*

INTRODUCTION

Industrial process requires large quantity of energy and raw materials, which release by-products such as liquid effluents, emission of CO_2 or other greenhouse gases during each and every step of process. CO_2 is one of the greenhouse gases that trap the radiation and more specifically infrared radiation coming from sun and prevents it from escaping. This is useful in maintaining the warmth on earth which has made earth suitable for life. However, excess trapping of heat causes the earth to get excessively warm which is undesirable.[1] These effluents and emissions have a very adverse impact on the environment. CO_2 contributes over 60% to global warming as a Greenhouse gas due to its huge emission amount.[2] A European report stated the estimated quantity of CO_2 emitted into the atmosphere approximately to be 34 billion tonnes in 2010[3], which increased to more than 36 billion tonnes in 2019[4] and these emissions are stated to increase over the next few decades. This significant growth in emissions can be attributed to increased human industrial activities such as thermoelectric power plants, cement plants and steel plants.[5] The CO2 concentration in atmosphere is now close to 400 ppm which is significantly higher than the pre-industrial level of about 300 ppm.[6]



Graph 1 Line Chart displaying increment in global CO_2 emissions[4]

This has caused problems with weather patterns and other geochemical cycles around the world and, according to the fifth assessment report from the Intergovernmental Panel on Climate Change (IPCC), strong action is required to maintain a sustainable living environment for the future generation.[7]

Many attempts have been made to reduce the CO_2 emissions from human activities to counteract the undesirable effects of climate change. Carbon capture and storage (CCS) technologies are engineering solutions which target the capture of CO2 that is produced from major emitters such as fossil fuel-fired power plants. This carbon dioxide can then be stored underground in geological formations such as deep saline aquifers or disused hydrocarbon reservoirs, or precipitated as carbonates that are environmentally inert.[8,9] Furthermore, various technologies and methods are under investigation to convert this CO_2 into some value-added products, which will prove quite useful considering the current scenario. One such process is the conversion of CO_2 into H_2CO_3, using microbial CA.

What is Carbonic Anhydrase?

Carbonic Anhydrase is a metalloenzyme having zinc metal in its active centre. It is a metalloenzyme that regulates important biological processes within humans and other living organisms such as the acid–base balance within the blood,[10] the photosynthesis mechanism in plants and the carbon concentration mechanism in microorganisms.[11] Apart from these processes it is used in many other physiological processes such as respiration, bone resorption, calcification and photosynthesis. Essentially we can say that it is ubiquitously found in all the kingdoms of life.[12] It is an enzyme that assists in rapid conversion of CO_2 and H_2O into carbonic acid and bicarbonates.

Carbonic Anhydrase was first identified in 1933, in red blood cells of cows. Since then, it has been found to be in abundance in all plants, mammalian tissues, bacteria and algae. This ancient enzyme has five distinct classes (called alpha (α), beta (β), gamma (γ), delta (δ) and zeta (ζ) carbonic anhydrase)[13,14]. Members of these different classes

have very little in common be it structurally or sequentially, yet they all perform the same function and require a metal ion at the active site.

The most commonly studied among these is the α-carbonic anhydrase[15] which is generally found throughout the animal kingdom.[16,17] In α-CA, the enzyme activity is due to a Zn^{2+} ion that is coordinated to three histidine residues near the centre of the molecule.[17]

The β-class is found in plants with C3 and C4 metabolism as well as monocotyledons, dicotyledons, arthropods and bacteria.[17] In this case, the Zn^{2+} ion is coordinated to two cysteine residues and one histidine residue on the protein.[18]

The γ-class is found in eubacteria.[19] It is considered as a perfect enzyme since it performs the reaction as fast as the carbon dioxide molecules can diffuse to it.[20]

ζ-CA, is found in diatoms and has Cd^{2+} as the metal ion catalyst instead of Zn^{2+},which reflects the lack of Zn^{2+} available in a marine environment.[21] However, another inference can be taken out from this is that ζ-CA is more efficient at CO_2 hydration when Cd^{2+} ion is replaced with Zn^{2+}.[22]

The δ-CA is another class of Carbonic anhydrase that is found in oceanic creatures such as dinoflagellates and phytoplankton. This combined with the photosynthetic ability of the oceanic organisms make Oceans the greatest sink of CO_2.[23]

α-CA is always present as a monomer, β-CA exists as dimers, tetramers, hexamers, and octamers whereas γ-CA is mostly present in single cell micro-organisms and always exists as a trimer.[18]

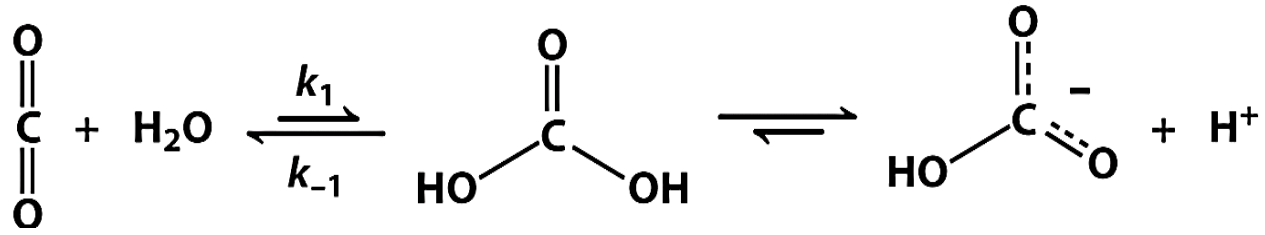


Figure 1. Reaction which is catalysed by Carbonic Anhydrase[24]

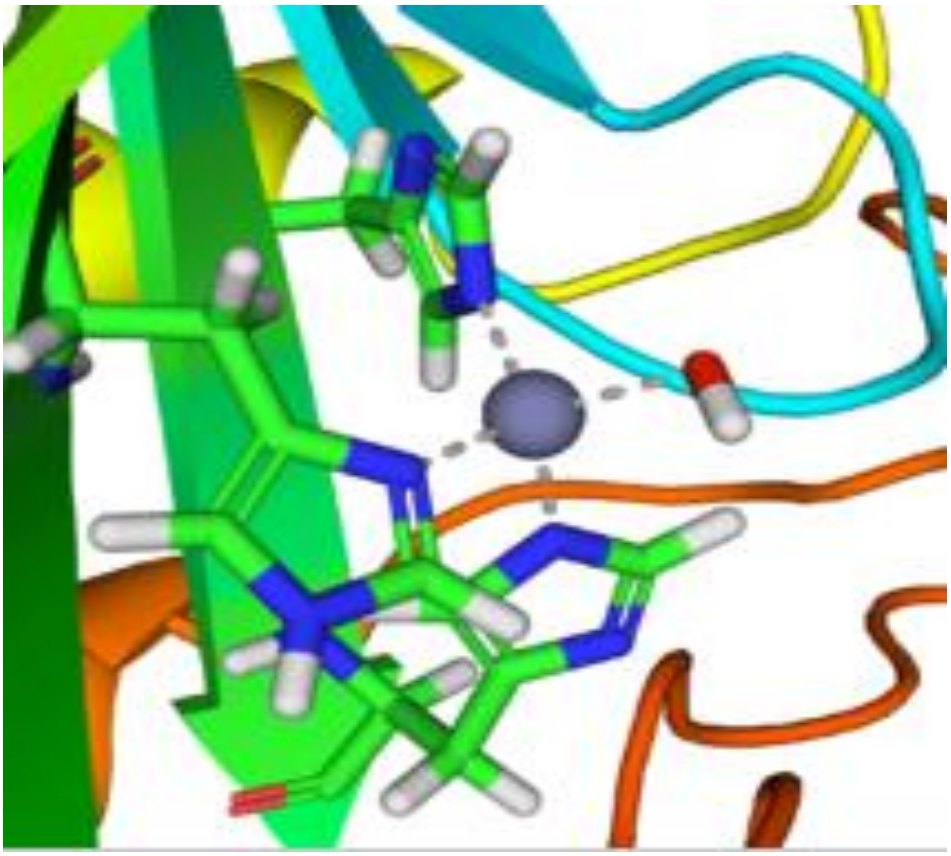


Figure 2. Human Carbonic Anhydrase II, showing 3 histidine residues and a hydroxide group coordinating zinc at its active center

Parameters used in measuring Enzyme Efficiency

- **K_{cat}:** is the turnover number. It tells us about the number of times each enzyme site converts substrate to product per unit time.[25]
- **K_m:** It is the Michaelis constant that is the substrate concentration needed to achieve a half maximum enzyme velocity. It is the concentration of substrates when the reaction reaches half of V_{max}. A small K_m indicates high affinity since it means the reaction can reach half of V_{max} in a small amount of substrate concentration. V_{max} represent the maximum rate achieved by a system, at maximum substrate concentration.[25]

- **K_{cat} /K_m ratio**: Increasing the reaction rate of a chemical reaction allows the reaction to become more efficient and hence more products are generated at a faster rate. This is known as catalytic efficiency of an enzyme.(mol/s).The higher the K_{cat} is, the more substrates get turned over in one second.[25]

- **Temperature:** Temperature plays a very important role in any biochemical system. This is because even a slight variation of temperature can have drastic change in results. Enzymes also operate in very narrow temperature range and hence this is an important parameter.

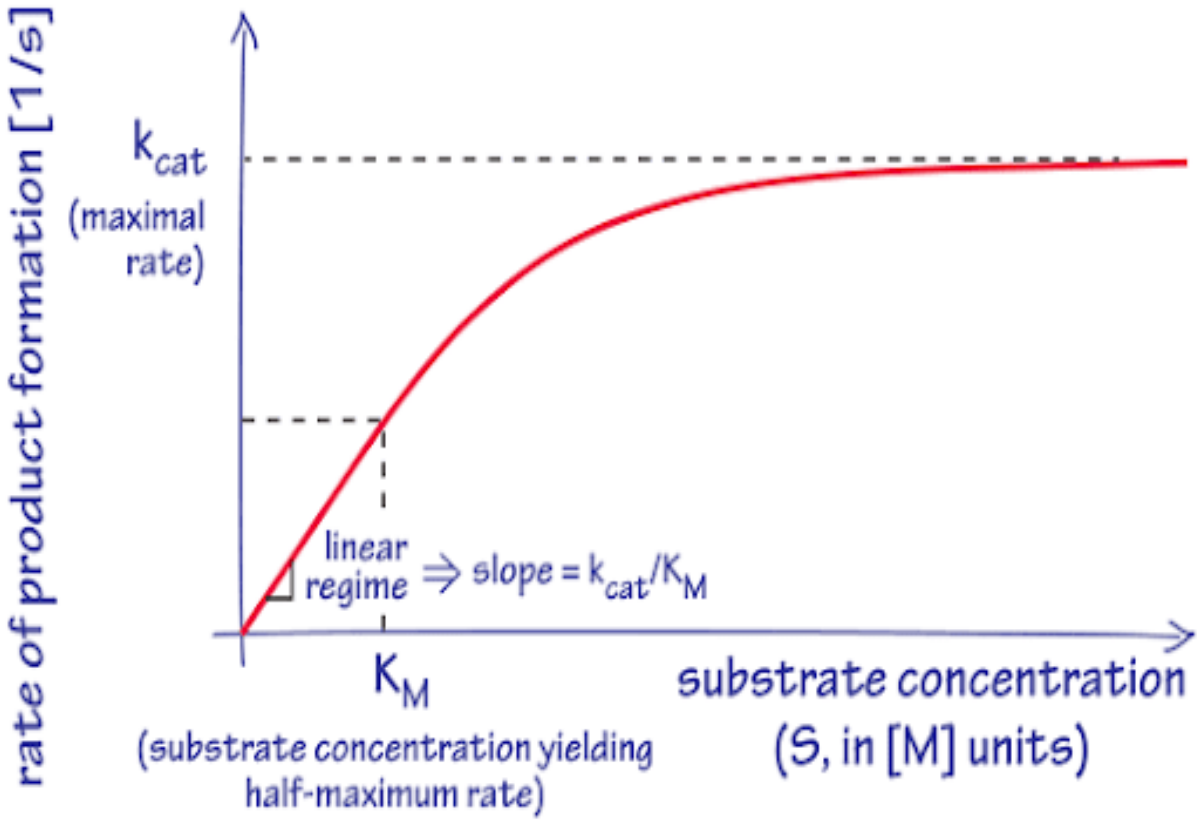


Figure 3. Graphical representation of enzyme parameters

Mechanism of working of Carbonic Anhydrase

All the classes of Carbonic anhydrase work in more or less same way despite having differences in their structure. The diagram explains the working in detail graphically.

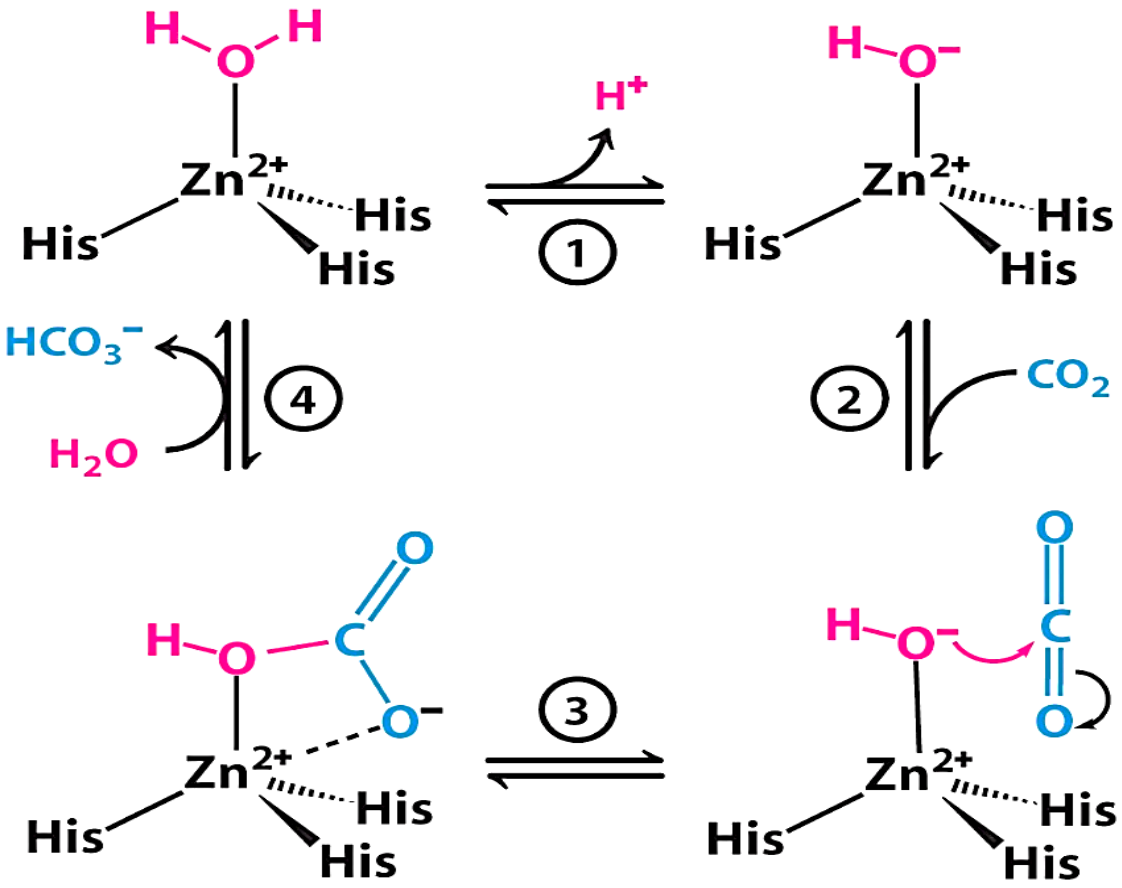


Figure 4. Reaction Mechanism of Carbonic Anhydrase[24]

Major Pollutants - Industries

A substantial number of industries be it small scale or large scale have carbon footprint and they pollute the environment in some way or the other. In spite of this the contribution of industries in reducing this is not so significant as most of the industries do almost nothing in order to reduce this carbon and greenhouse gas footprint created by them. Larger industries do employ some techniques like afforestation as a part of their Corporate Social Responsibility (CSR) Program. But this is not enough. Transport, Energy and Textiles are some of the most polluting industries in India. In 2014 Central Pollution Control Board (CPCB) directed state pollution control bodies in India to Install Continuous Emission Monitoring Systems in most polluting industries throughout India. Regardless of this, as many as 18% of the industries did

not comply to this.[26] This has made pollution an even serious issue in present times, since people are already suffering from a lot of airborne diseases like asthma and bronchitis. Need of the hour is to stop this since patients of these diseases are on the rise and would continue to increase by time. Bad quality air would only act as a catalyst to this.

However, we cannot stop or close down industries due to this reason. This is because if we do so, the economy would collapse. But we definitely need to come up with some solution to this. Carbonic anhydrase can come to our rescue in this. Carbonic Anhydrase is used in living organisms in respiration process by converting CO_2 into Carbonic Acid. Similar to this, it can be used in industries. This converted carbonic acid, carbonates and other form of captured CO_2 would further be useful again in the industries and promote for a circular economy.

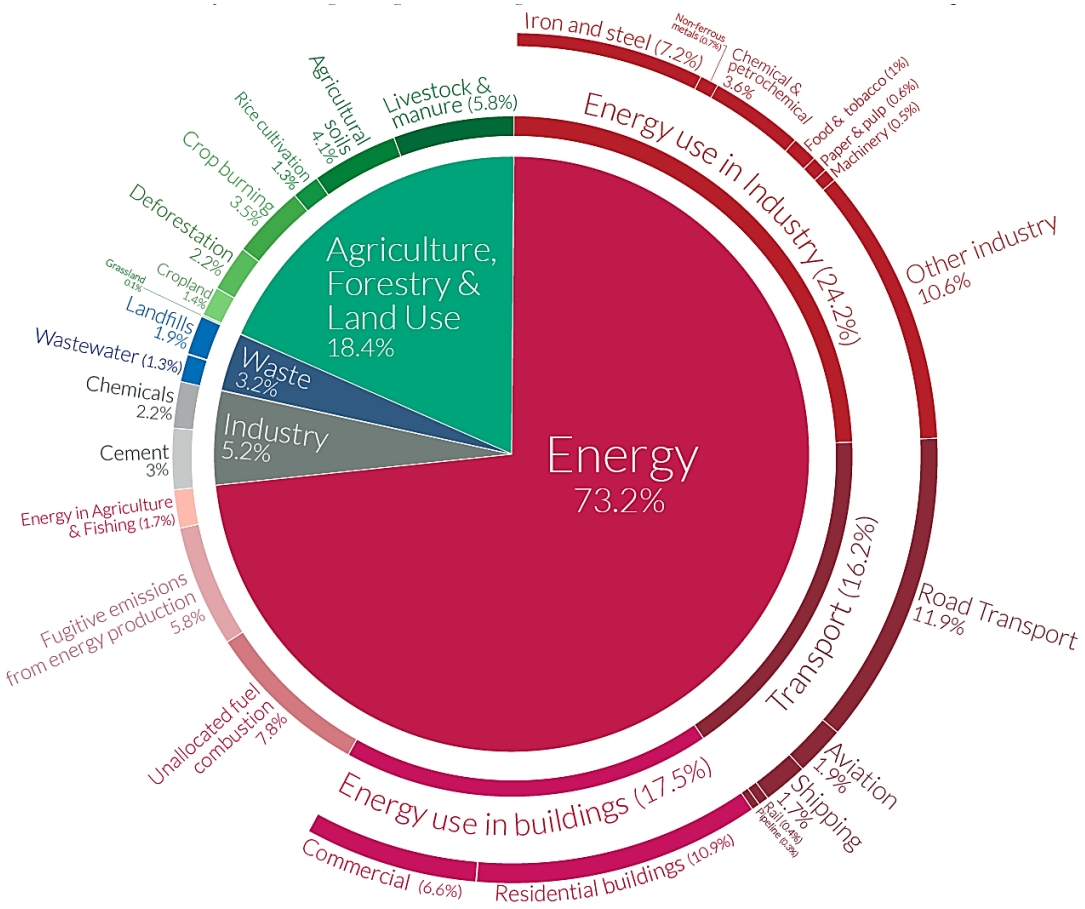


Figure 5 Share of different sectors in total CO_2 emission[27]

Circular Economy – Concept

Circular economy is a new concept in modern era. Presently Industries use the model of take – make – dispose, which is they take raw material, make useful goods from it and throw away or dispose off the waste. This waste with little to no recycling can be useful to the same industry or to a different industry. If this processing is done, we can reduce wastage and increase profits from minimal products and raw materials. A circular economy aims to redefine growth. It tends to gradually decouple economic activities from the finite resources we have and then making waste as less as possible. It also focuses on the use of renewables, be it energy or materials as much as we can. Basically, circular economy is inspired from the natural and biological cycles of the nature which, if we see a bigger picture, does not waste anything. Everything is used somewhere or the other.[28]

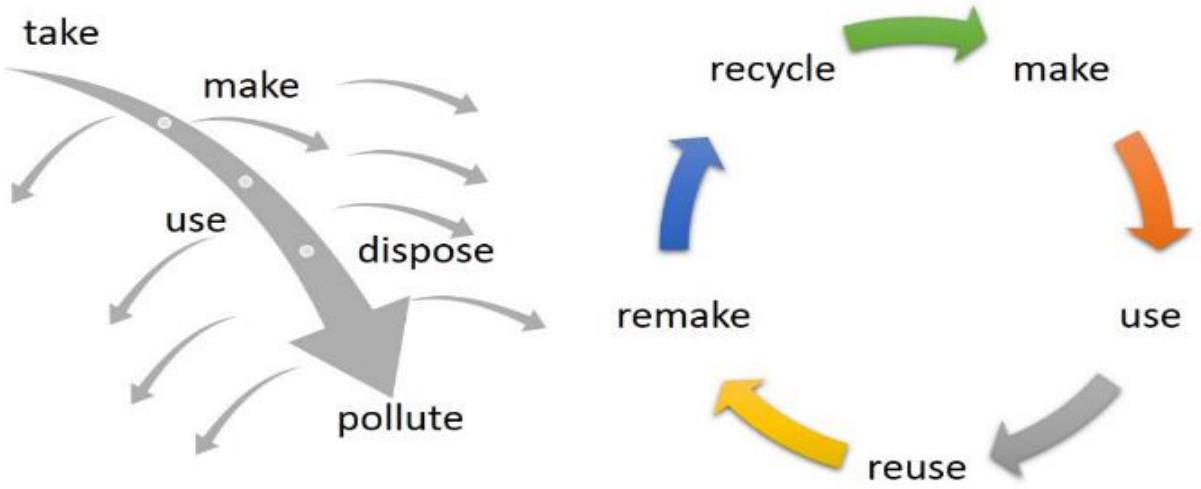


Figure 6. Linear vs Circular Economy Concept[29]

As we can see in the figure above, there is a whole lot of difference in the approach taken by Linear or traditional economic principles and that of circular economy. Linear principles involve a lot of wastage at each and every step of production. If we look at it collectively, we can notice that there is a lot of resources that remain unused and discarded just because we do not try to reduce, reduce or recycle them. However, circular economic principles are completely opposite from this in their approach. In

this, we try to maximise utilization of each and every resource. It is achieved by reducing the usage of resources by optimizing the process in general; then reusing the once used resources in either the same industry or by some other industry; and lastly recycling is done to recover loss of materials from depleted sources so as to maximise it usage.

So basically, there are two ways in which captured CO_2 can be used in the industries. One is the direct utilization of CO_2, that is used in various industries which require CO_2 as a raw material and second is the indirect usage of CO_2 by converting it into energy products, chemicals and various other industrial materials.[30]

CO_2 is used directly in various industries such as in soft drinks manufacture, welding, foaming, propellants as well as CO_2 at supercritical temperatures is used as solvent. CO_2 capture via photosynthesis to directly fix carbon into microalgae has also attracted attention of researchers.[30]

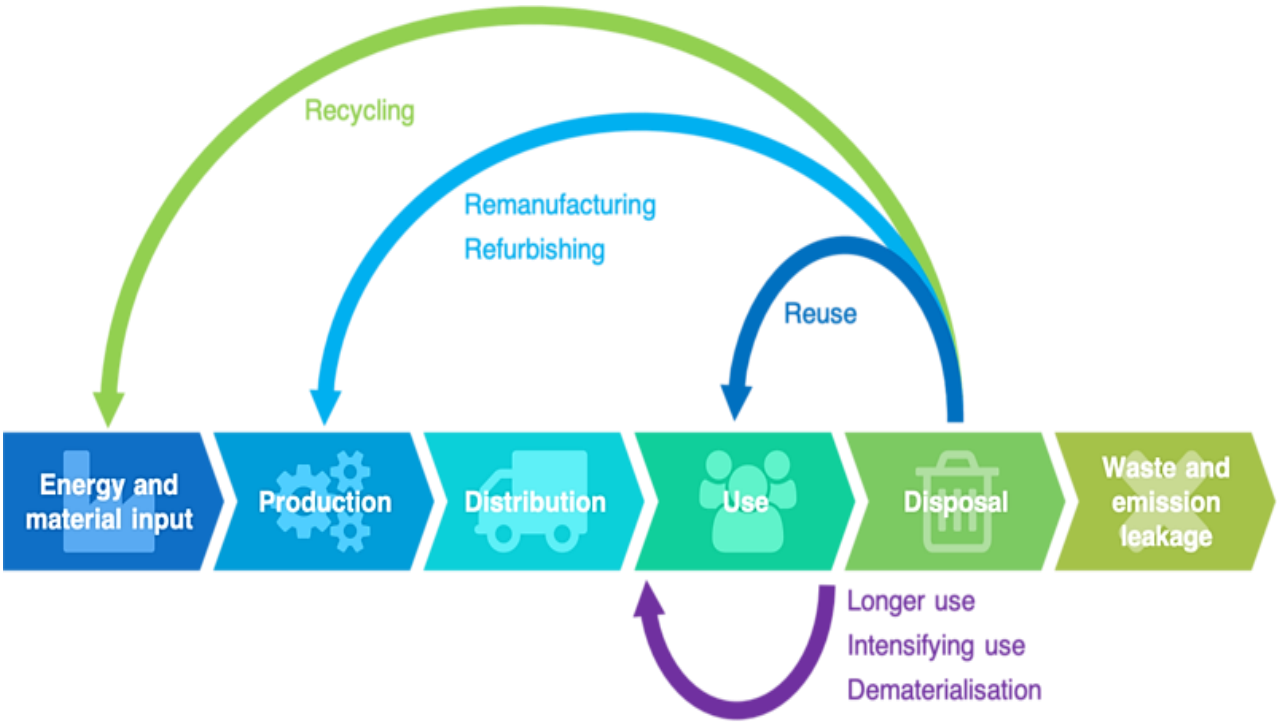


Figure 7. The Circular Economy Concept[31]

Major Uses of Captured CO_2

There are various uses where the CO2 captured from various industries find a significant amount of application. Some major use cases include

- The Conversion of CO_2 into methanol. This is a very important process for the use by industry as the process of conversion of CO_2 into alcohol can be cheaper as compared to from biomass, and it is less inflammable. The methanol produced by this can further be utilized as a clean-burning fuel, in pharmaceutical industries, automobiles industries and as a general solvent in various industries.[32] This would be cheaper for industries and solve a major environmental issue.

- Conversion of CO2 into industrially important biopolymers. Polyhydroxyalkanoate (PHA) and Polyhydroxybutyrate (PHB) biopolymers are widely recognised as outstanding candidates to replace conventional petroleum-derived polymers. Their mechanical properties are good and can be customized through copolymer composition, they are biodegradable, and unlike many alternatives, they do not rely on oil-based feedstocks. Further, they are the only commodity polymer that can be synthesised intracellularly, ensuring stereoregularity and high molecular weight.[33]

- The stored carbon can be put in underground geological formations. This would basically increase the rate of fossil fuels formation and then it can again be used as fuels. Not only this would reduce the carbon dioxide from the atmosphere but also help in increasing our fossil fuel storage. This is as close as we can get in converting fossil fuels that are considered as non-renewable resources to somewhat renewable resources.[34]

As of 2017, there were almost two dozen commercial scale carbon capture and conversion projects going on and were almost twenty-two in development stage. This itself speaks volumes about the awareness towards carbon capture and sequestration and it's potential.

If this model is adopted by our industries, this would mean a sustainable and a much more efficient process for the industry and this would lead to industries making much more in profits and environment still being habitable by humans.

Review of Literature

CA have led the drive for an alternative fuel supply and have enabled the development of an efficient and inexpensive carbon sequestration system [35]. Various studies have been directed at improving the methods of protein engineering for CA to overcome the problem of low stability and long operation time across a wide range of temperature and pH. Burton et al developed a computational algorithm and predicted potential cysteine residue pairs for disulfide engineering to stabilize human CA II.[36] Moreover, Barbero et al built engineered yeast (Saccharomyces cerevisiae) cells displaying recombinant CA or mineralization peptides for enhanced CO_2 hydration, $CaCO_3$ mineralization and particle settling rate.[37] Furthermore, in a study from Jo et al. 2013, CA from Neisseria gonorrhoeae in periplasm of E.coli was highly expressed in soluble form.[38] Undoubtedly, thermostability significantly affects the application of CA in carbon sequestration system. However, this process increases the cost of CA and it is very difficult to scale up to an industrial level. Therefore, a new CA with high stability and activity is highly desirable for carbon sequestration system. We reviewed the following papers majorly to get the content that is reported in this report.

S.No.	TITLE	AUTHOR	PUBLISHED DATE	REMARKS	JOURNAL
1	Efficient reduction of CO_2 using a novel CA producing *Corynebacterium flavescens.*	Tanvi Sharma, Ashok Kumar	08-Jul-20	The study was aimed at isolating CA producing bacteria from cow saliva. The crude enzyme was tested for the conversion of CO_2 into the calcium carbonate ($CaCO_3$) under controlled conditions.	Korean Society of Environmental Engineers
2	Usefulness of Kinetic Enzyme Parameters in Biotechnological Practice.	Néstor Carrillo , Eduardo A. Ceccarelli & Oscar A. Roveri	15-Apr-13	This study provides general formalisms to calculate enzymatic parameters that could be applied to real industrial conditions.	Biotechnology and Genetic Engineering Reviews
3	CA: An Efficient Enzyme with Possible Global Implications	Christopher D. Boone, Sonika Gill, Andrew Habibzadegan, and Robert McKenna	19-Aug-13	This review highlights some of the accomplishments of CA in the reduction of CO_2	International Journal of Chemical Engineering

Table 1 Lliterature having major impact on this study

MATERIALS AND METHODS

Industries use numerous methods to reduce the amount of GHGs released by them and in the same process they try to obtain carbon credits for this. The processes range from simple mechanical processes like filtration and adsorption to advanced chemical processes like scrubbing. However, the most efficient processes that tend to capture and converts CO_2 into useful products are biological processes. This we can see in nature also. The natural way of replenishing the CO_2 is by plants which use it as a carbon source for their energy synthesis. This is one of the most efficient carbon capture and conversion process since apart from converting CO_2 into energy source for plants, it also replenishes the O_2 in the environment. Additionally, plants also need other inorganic substances and hence they also convert and reduce other GHGs in the environment. This is also the main reason why afforestation is said to help a lot in reducing the GHGs in the environment.[39]

However, industries have limited space to plant trees within their premises and the amount of afforestation that can be done by industries would help little in restoring the carbon emitted by them. In this scenario using the enzymatic processes to convert CO_2 into other industrially useful products can be very helpful to the industry.

Why Carbonic Anhydrase?

- In absence of enzyme, this reaction is very slow, with 200 molecules of H_2CO_3 being formed in an hour. In presence of enzyme the reaction speeds up dramatically with about 6000,000 molecules formed every second i.e. the enzyme has accelerated up the reaction by 10 million times. [40]
- CAs are involved in diverse physiological functions including pH regulation, ion transport, bone resorption and secretion of gastric, cerebrospinal fluid and pancreatic juices.[40]
- The most important function of CA is related to the respiration and transport of CO2/bicarbonate in various metabolizing tissues. [40]

- This enzyme is also involved in electrolyte secretion, CO_2 and pH homeostasis, CO_2 fixation and biosynthetic reactions such as gluco-neogenesis and ureagenesis.[40]

In present study, a statistical model is developed, that would compare and recommend the carbonic anhydrase to be used in different industrial processes based on their temperature requirement. The K_{cat} /K_m ratio is used, for selecting a CA producing organism having higher efficiency for conversion of substrate into products. In this study, *Sulfurihydrogenibium azorense,* a bacterium was chosen as a model organism with highest catalytic efficiency of 350000 ($1/mMs^{-1}$). Similarly, several organisms were selected which grow on variety of temperature (thermophilic, mesophilic, psychrophilic) and their efficiency was compared at different temperature. Using statistics and, graphical analysis the data was plotted and a model was developed that compares and recommends the carbonic anhydrase that can be utilized in various industrial processes based on their temperature, pH etc.

We have developed a computer program for predicting the best suited CA so as to make the task somewhat easier and streamlined for the industries.

This program is based on database matching and statistics. It takes in the input in the form of temperature at which we need the carbon capture to be happening.

Our Application

For this purpose, we need to develop a software which can easily determine the carbonic anhydrase to be used in an industrial process to capture and convert the CO_2. The software that we aim to develop is based on Node.js programming language. This is due to the fact that we can make software independent of operating systems like Windows, Mac or Linux using Node.js. In this what we basically develop are Web-applications and then compile them to make native and independent software packages. These are able to install and run on any computer that the user needs.

What is Node.js?

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. It was initially written by Ryan Dahl back in 2009. The initial release only supported Unix based systems like the ones based on Linux and MacOS and support for Windows was added later to it.[41,42] It basically is a high-level programming language just like C, C++ and Python. Node.js lets developers use JavaScript to write command line tools and for server-side scripting that is running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript is everywhere" paradigm,[43] which tries to unify web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

V8 is the JavaScript execution engine which was initially built for Google Chrome, the world's most used web-browser. It was then open-sourced by Google in 2008. V8 compiles JavaScript source code to native machine code at runtime.[43] It's job is to read the high level code that is written by the user and to translate it into assembly and then binary code so as to make it computer usable.

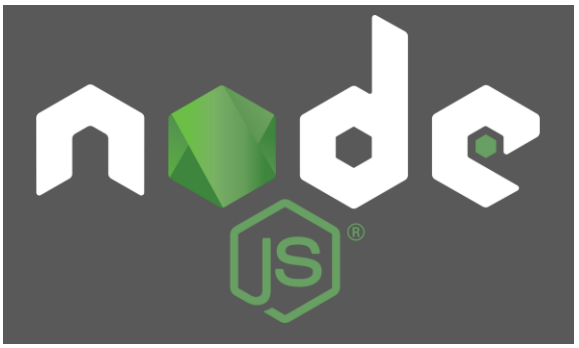


Figure 8 Node.js logo

'npm' or Node Package Manager is the pre-installed package manager for the Node.js platform. It installs Node.js programs from the npm registry, organizing the installation and management of third-party Node.js programs. Packages in the npm registry can

range from simple helper libraries such as Lodash to task runners such as Grunt. It has non-exhaustive list of applications and packages available for using it in different projects to suit the need of the user.

Node.js registers with the operating system so the OS notifies it of connections and issues a callback. Within the Node.js runtime, each connection is a small heap allocation. Traditionally, relatively heavyweight OS processes or threads handled each connection. Node.js uses an event loop for scalability, instead of processes or threads. In contrast to other event-driven servers, Node.js's event loop does not need to be called explicitly. Instead, callbacks are defined, and the server automatically enters the event loop at the end of the callback definition. Node.js exits the event loop when there are no further callbacks to be performed.[44]

In our application, we aim to make a tool using Node.js which based on statistics and structural data of CA, which helps to determine the CA which should be used for a particular industrial process. This would help the industries easily decide the organism and CA which must be used so as to deploy the CO_2 capture and sequestration system.

Initially we aim to make it for local machines which the users can download as a package that is precompiled for MacOS, Windows and Linux according to their needs. Further, we can also make it source code available so that it can be compiled for different other systems based on the Operating systems and Architecture (x86, x64, ARM) and so on.

Later on, we can make it available as a web-server since it is based on Node,js which can easily be deployed on web servers, which would give it more flexibility and scalability. This is due to the fact that web-servers can be run even when the user machine is not powerful since calculations can be easily done on powerful web servers, and large databases can be maintained on web servers. As the database of organisms and enzymes grows, we will need to get it on web server since on servers database can be updated at a single place and users can have access to updated database seamlessly.

Parts of the Program

This program is developed on node.js which is basically a web application developing software, however GUI based standalone applications can also be quite easily developed using it. The backend of the program has the following parts:

i. index.html

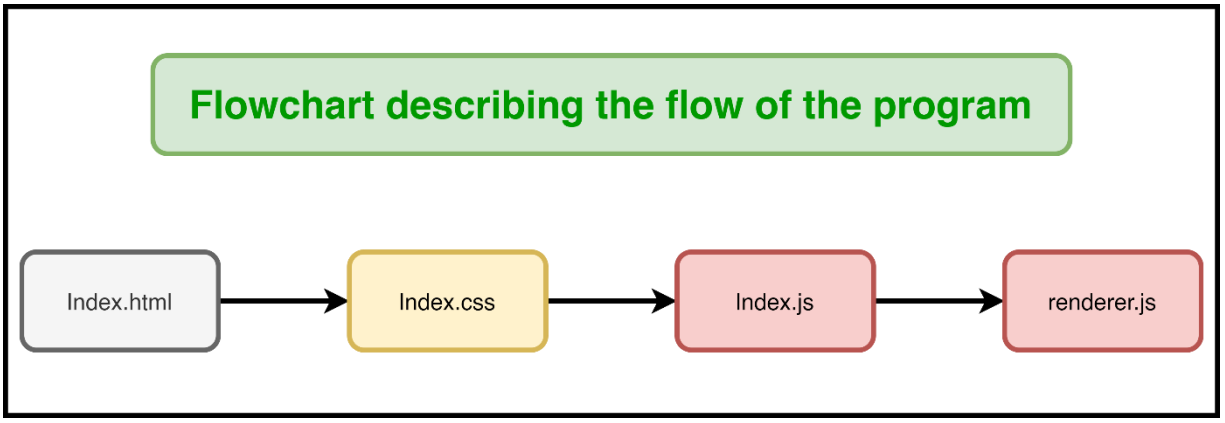
ii. index.css

iii. index.js

iv. renderer.js

v. data_plot.json

Flowchart of the application process



Along with this, there is the data-plot.json file which is the database file and is independent of the logic of the program. It is called upon by renderer.js when data is queried and processed in the program.

These all interconnect together and run it seamlessly for a good user experience. All the files which collectively make the program along with their code is in the appendix section.

Along with this, there are 3 files which are not the part of the application, but are necessary for the working since these are the files required to compile the application. These are files that are part of the node.js package and electron application package. These files are:

i. forge.config.js

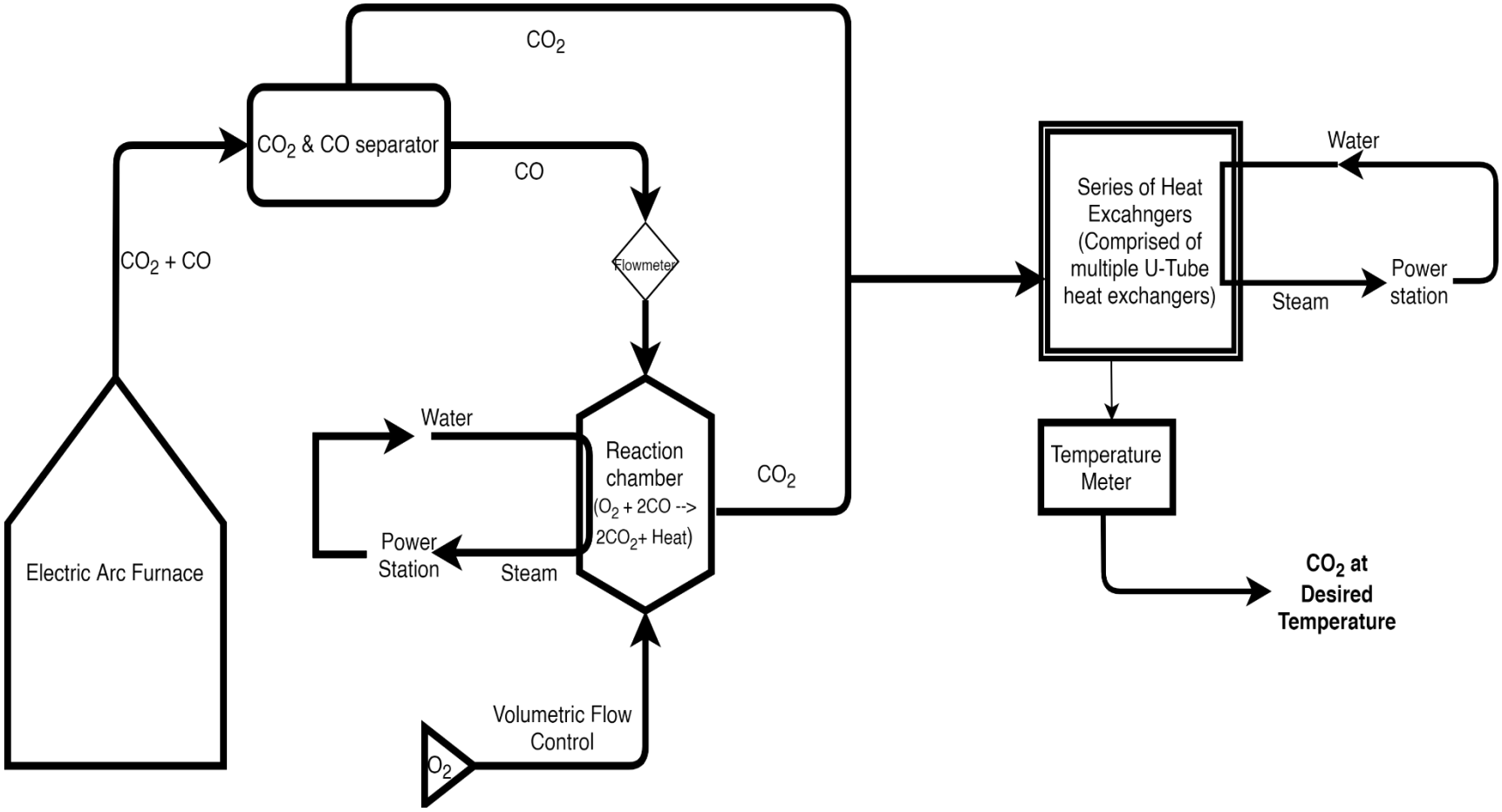
ii. package.json

iii. package-lock.json

The Process

We have made structural diagram by taking the example of a ferro-manganese alloy industry. It is an industry which is highly linked to the progress of the nation since it is required in production of steel. In this we have made a rough outline of processes that can be again run from the by-products of the main process that is the production of Mn-Fe alloy.[45]

Process Diagram of Fe-Mn Alloy Production



Steps involved in production of Fe-Mn Alloy:

- First the reaction is made to take place in an electric arc furnace. In this process, the alloy is produced and subsequently GHGs are released in the process. A major chunk of these GHGs are CO_2 and CO.
- In existing setups also, a number of industries use the separator to separate out CO and CO_2. This is so that CO_2 can be released as it is since it is way less lethal than CO.
- CO is then again heated and made to oxidise with added oxygen so as to convert it into CO_2. The heat generated by this is used to generate electricity. This is again useful for the industry.
- Then the CO_2 is released from this.
- We intend to collect CO_2 from both the outlets and since it is at high temperature, route it to a series of U-tube heat exchangers, which can cool this down. Every outlet of heat exchanger has thermometer to detect the temperature of outcoming gas. This can be set to automatically release the gas to a set outlet once it has reached a specific temperature needed by us.
- The energy released in the form of heat from this can also be used to produce electricity, by the same way done earlier.
- And the CO_2 released can be taken up to process it in Carbon sequestration process by the organism that we have.
- These organisms have CA which captures and converts CO_2 into bicarbonates and carbonic acid, which are again useful for industries.

The enzymes need a specific set of conditions which are very important for them to work efficiently. We have attached a graph of temperature with the activity of enzymes which we have collected. These can be broadly classified as the operating conditions of the enzymes and from the organisms they are isolated.

Moreover, we cannot use each and every organism directly. Some organisms cannot be cultivated easily in the lab and we need to take the genomic sequence of these organism to engineer their genome which produces the enzyme into some easily cultivable bacteria like *E. coli*. In this we also need to identify the active site of enzymes and compare them so as to get the idea why a certain enzyme works best and not the other.

Finally, when we have all the data, we need to make a computational model of this and using regression & interpolation, we need to find out which organism is best in what temperature range.

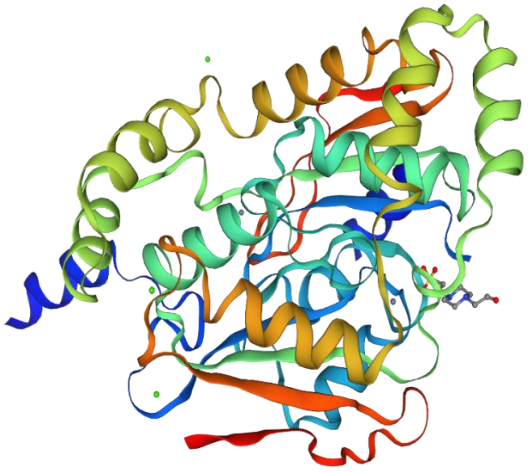
However, at present we have a very limited dataset and we cannot make an accurate model using such a limited dataset, hence we have just made a software that can reliably predict the CA based on temperature as a parameter.

DATA COLLECTED

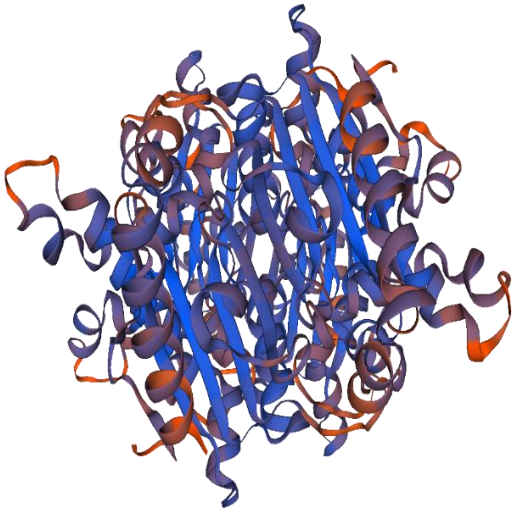
During the course, we have collected the following data.

Enzyme Structures

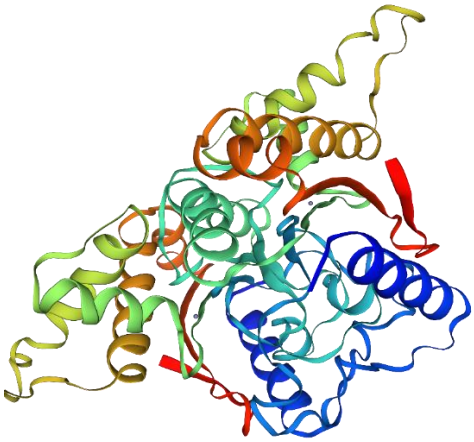
Ribbon structures of CA enzyme found in different organisms.



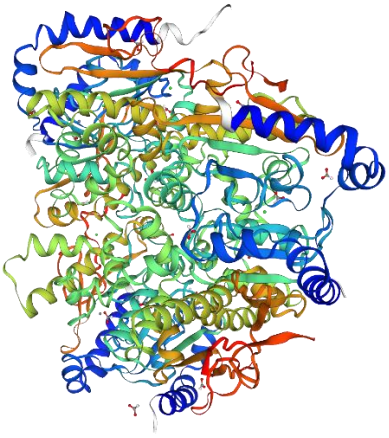
(a) Methanothermobacter thermautotrophicus



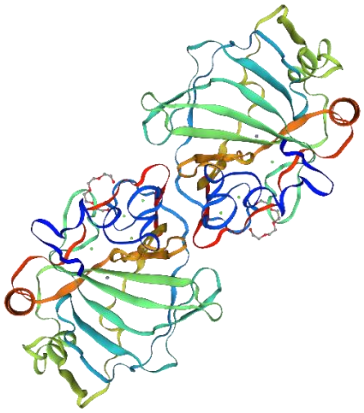
(b) Brucella suis



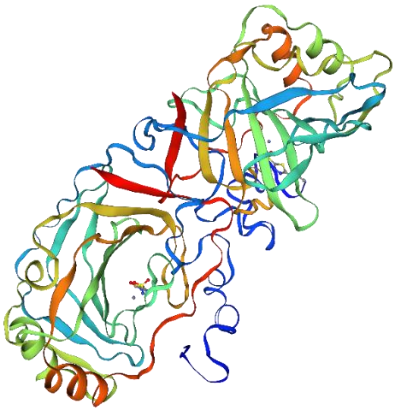
(c) Porphyromonas gingivalis



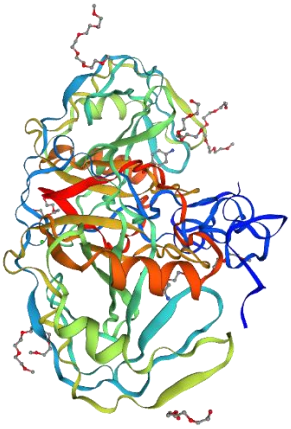
(d) Synechocystis sp. PCC 6803



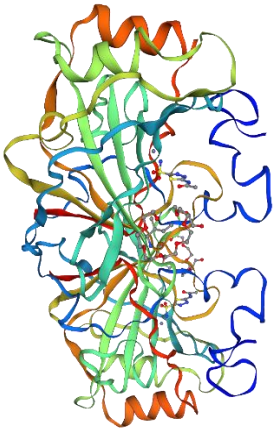
(e) Persephonella marina



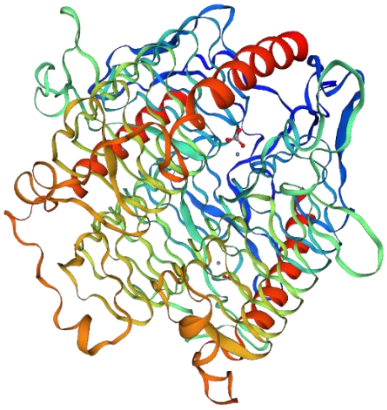
(f) Sulfurihydrogenibium sp. YO3AOP1



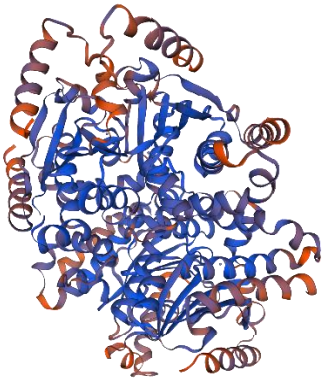
(g) Thermovibrio ammonificans



(h) Sulfurihydrogenibium azorense



(i) Methanosarcina thermophilia

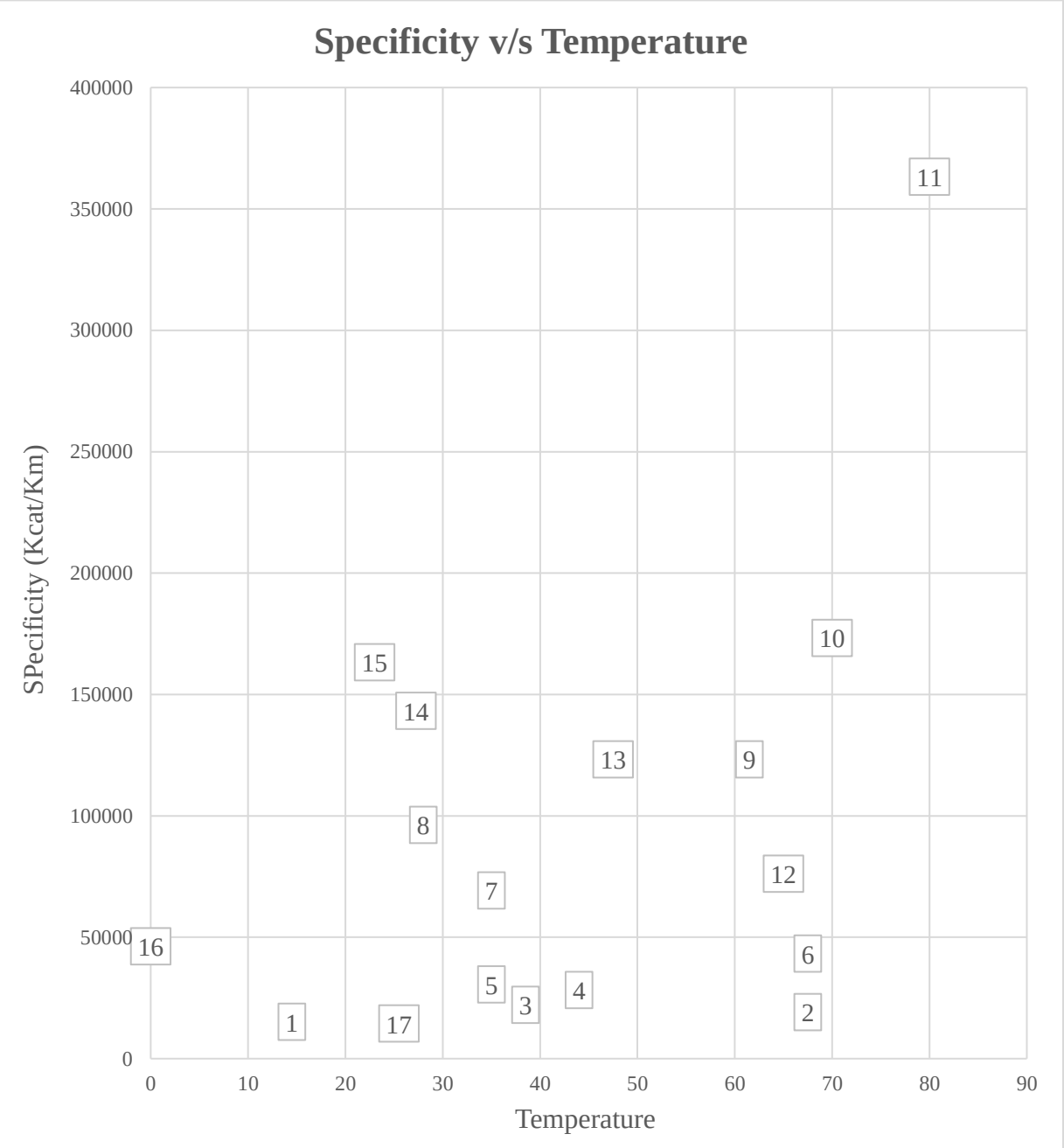


(j) Pseudoalteromonas translucida

Figure 9 CA enzymes from different organisms

Optimum Temperature v/s Specificity Graph

Graph depicting a scatter of optimum temperature v/s specificity of CA from different organisms.



Graph 2 Graph Depicting optimum temperature of different CAs

- Please refer to the corresponding numbers against data in table below.

S. No.	Temperature Range	Category
1	Up to 10 ℃	Psychrophilic
2	10℃ - 60℃	Mesophilic
3	Greater than 60℃	Thermophilic

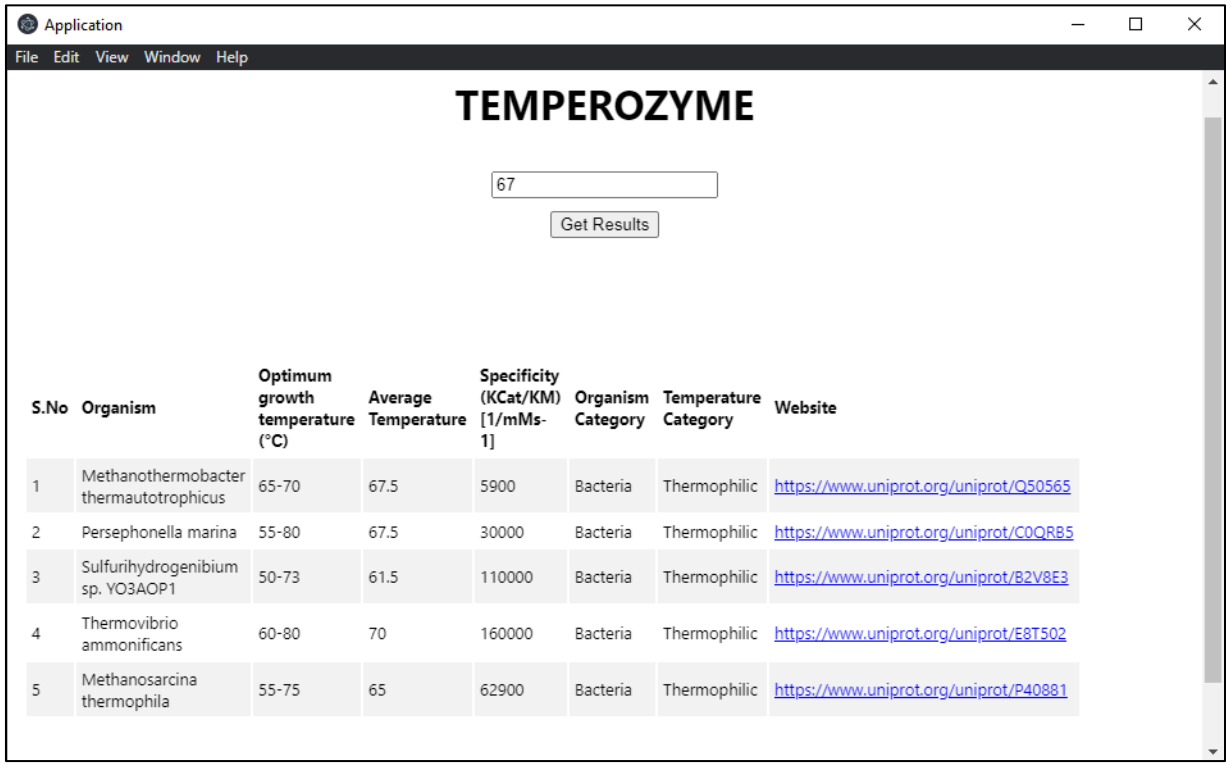
Table 2. Temperature wise categorization of organisms[46]

S. No	Organism	T_{opt} (℃)	T_{avg} (℃)	K_{Cat}/K_m	Category
1	*Pseudoalteromonas translucida*[47]	*0-29*	14.5	*1900*	*Bacteria*
2	*Methanothermobacter thermautotrophicus*[48]	*65-70*	67.5	*5900*	*Bacteria*
3	*Brucella suis*[49]	*37-40*	38.5	*8900*	*Bacteria*
4	*Porphyromonas gingivalis*[50]	*41-47*	44	*15000*	*Bacteria*
5	*Synechocystis sp. PCC 6803*[51]	*32-38*	35	*17300*	*Bacteria*
6	*Persephonella marina*[52]	*55-80*	67.5	*30000*	*Bacteria*
7	*Enterobacter hormaechei*[53]	*35*	35	*56000*	*Bacteria*
8	*Nostoc commune*[54]	*28*	28	*83000*	*Bacteria*
9	*Sulfurihydrogenibium sp. YO3AOP1*[55]	*50-73*	61.5	*110000*	*Bacteria*
10	*Thermovibrio ammonificans*[56]	*60-80*	70	*160000*	*Bacteria*
11	*Sulfurihydrogenibium azorense*[57]	*80*	80	*350000*	*Bacteria*
12	*Methanosarcina thermophila*[58]	*55-75*	65	*62900*	*Bacteria*
13	*Columba livia*[59]	*45-50*	47.5	*110000*	*Animal*
14	*Danio rerio*[60]	*26-28.5*	27.25	*130000*	*Animal*
15	*Homo sapiens*[61]	*20-26*	23	*150000*	*Animal*
16	*Thalassiosira weissflogii*[62]	*0*	0	*33000*	*Algae*
17	*Sordaria macrospora*[63]	*15-36*	25.5	*1210*	*Fungus*

Table 3. Database of organisms used in application

- T_{opt} = Optimum Temperature for the growth of the organism
- T_{avg} = Mean value of the upper and lower limit of the optimum temperatures.

Figure 10. Hopmepage of the application



S.No	Organism	Optimum growth temperature (°C)	Average Temperature	Specificity (KCat/KM) [1/mMs-1]	Organism Category	Temperature Category	Website
1	Methanothermobacter thermautotrophicus	65-70	67.5	5900	Bacteria	Thermophilic	https://www.uniprot.org/uniprot/Q50565
2	Persephonella marina	55-80	67.5	30000	Bacteria	Thermophilic	https://www.uniprot.org/uniprot/C0QRB5
3	Sulfurihydrogenibium sp. YO3AOP1	50-73	61.5	110000	Bacteria	Thermophilic	https://www.uniprot.org/uniprot/B2V8E3
4	Thermovibrio ammonificans	60-80	70	160000	Bacteria	Thermophilic	https://www.uniprot.org/uniprot/E8T502
5	Methanosarcina thermophila	55-75	65	62900	Bacteria	Thermophilic	https://www.uniprot.org/uniprot/P40881

Figure 11. Application view when suitable CA organisms are found

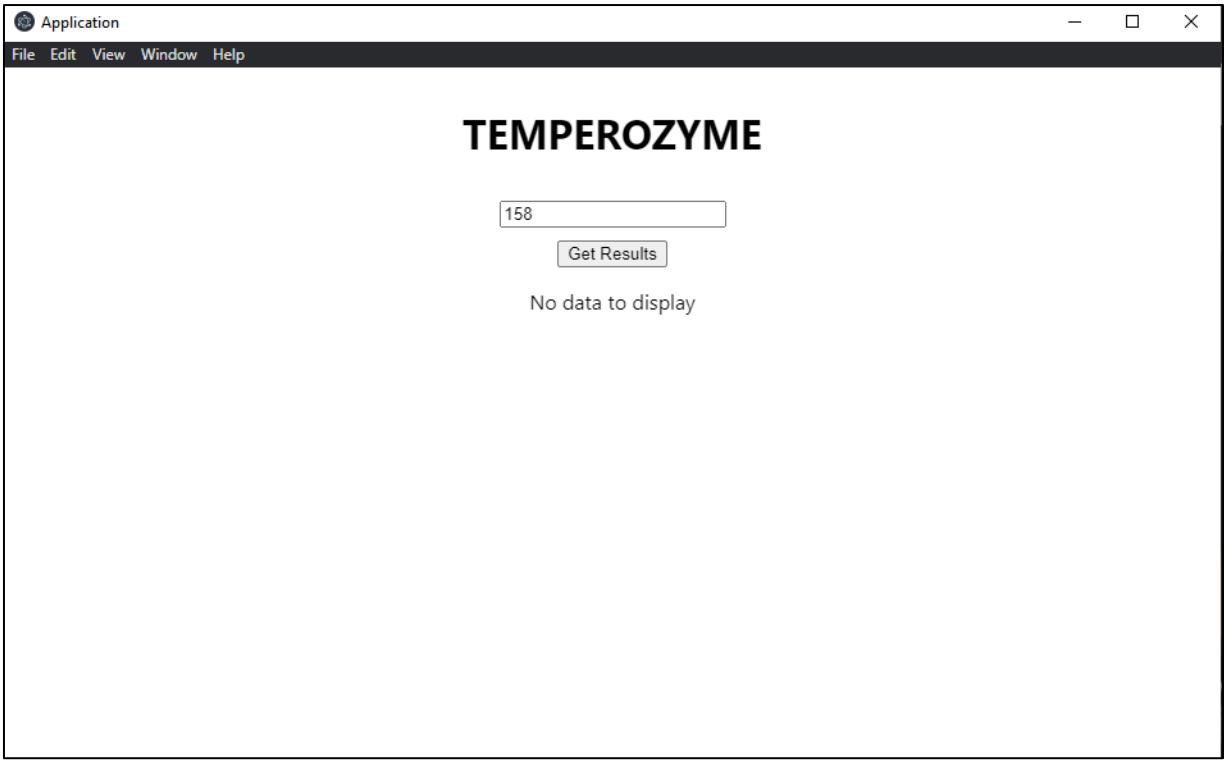


Figure 12. Application view when no suitable organism is found

APPLICATIONS

1. The Model which we aim to develop would have the data for enzymes at different parameters and could be used for various industrial processes just by changing those parameters.
2. This model would automatically suggest the organism from whose CA is best suitable. Hence would save a lot of time for industries when installing CO2 sequestration units in the facilities.
3. Additionally, using the method would give out carbon credits which is good for the environment.

FUTURE WORK

We have collected a data of numerous organisms for the project that we aim to do. This data has been kept in a database file. With the help of this data, we aim to develop a model-based on statistics to find and recommend the best enzyme and organism for a certain temperature and other parameters. For this we have to create a regression model of the data obtained and try to fit the data in the model. After this, for the temperatures where there is no suitable perfect match, we can use interpolation techniques to find the best match for the said temperature range. This would help us in predicting the best suited organism or enzyme for carbon capture and sequestration. By only changing the temperature and other parameters which we expect to get from the effluent, by comparing from the model, we can easily predict the organism to use for the carbon capture and conversion process.

Additionally, we need to create graphs by taking other factors also into consideration such as activation energy. This could open doors to new models and would help us in creating a process very useful for the industries as well as making the environment much more habitable and freer of pollution.

Another advantage would be the promotion of circular economy. As we said earlier that this concept needs to be brought into our day-to-day life as well as production industries to minimise waste and maximise profits as well as for sustainability in manufacturing sector.

APPENDIX

The Application is based on node.js and is basically run by these 4 files. The code of each file is given below.

Index.html

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset="UTF-8">
      <title>Application</title>
      <link rel="stylesheet" href="index.css">
      <script src="renderer.js"></script>
   </head>
   <body>
    <div class="header">
      <h1>TEMPEROZYME</h1>
      <input type="number" id="inputTemperature"
onkeyup="isGetResultsButtonDisabled()" placeholder="Temperature
in Celsius"></input>
      <button id="getResultsButton" onclick="getResults()"
disabled="isGetResultsButtonDisabled()" >Get Results</button>
    </div>

 <p id="data-display"></p>

   </body>
</html>
```

Index.css

```
body {
  font-family: -apple-system, BlinkMacSystemFont, "Segoe UI",
Roboto, Helvetica, Arial, sans-serif;
  /* max-width: 38rem; */
  overflow-x: auto;
}
.header{
     display:flex;
     flex-direction: column;
     align-items: center;
     width:100%;
     gap:10px;
}
table {
  width:70%;
  margin:10% 5px;
  font-size: 0.8rem;
}
th, td {
  text-align: left;
  padding: 4px;
}

tr:nth-child(even){background-color: #f2f2f2}
```

Index.js

```
const { app, BrowserWindow } = require('electron');
const path = require('path');


// Handle creating/removing shortcuts on Windows when
installing/uninstalling.
if (require('electron-squirrel-startup')) { // eslint-disable-
line global-require
  app.quit();
}


const createWindow = () => {
  // Create the browser window.
  const mainWindow = new BrowserWindow({
    width: 800,
    height: 600,
    icon: __dirname + '/icon_128@2x.png',
    webPreferences: {
      nodeIntegration: true,
      nodeIntegrationInWorker: true,
      contextIsolation: false,
      enableRemoteModule: true,
  }
  });


  // and load the index.html of the app.
  mainWindow.loadFile(path.join(__dirname, 'index.html'));
```

```
  // Open the DevTools.
  // mainWindow.webContents.openDevTools();
};

// This method will be called when Electron has finished
// initialization and is ready to create browser windows.
// Some APIs can only be used after this event occurs.
app.on('ready', createWindow);

// Quit when all windows are closed, except on macOS. There,
it's common
// for applications and their menu bar to stay active until the
user quits
// explicitly with Cmd + Q.
app.on('window-all-closed', () => {
  if (process.platform !== 'darwin') {
    app.quit();
  }
});

app.on('activate', () => {
  // On OS X it's common to re-create a window in the app when
the
  // dock icon is clicked and there are no other windows open.
  if (BrowserWindow.getAllWindows().length === 0) {
    createWindow();
  }
});
```

Renderer.js

```
const fs = require('fs');

const path=require('path');

const shell = require('electron').shell


function isGetResultsButtonDisabled(){

    const getResultsButton =
document.getElementById('getResultsButton');


getResultsButton.disabled=(document.getElementById('inputTemper
ature').value==="");

    }


function readStore(){

    let rawdata = fs.readFileSync(path.resolve(__dirname,
'dataSources/data_plot.json'));

    let data = JSON.parse(rawdata);

    return data['For database']

}


function isTemperatureInRange(inputTemperature,range){

if(range.includes('-'))

{

    const splittedRange = range.split('-');

    if(inputTemperature>=Number(splittedRange[0]) &&
inputTemperature<=Number(splittedRange[1]) )

        return true;

    else return false;
```

```
}
if(inputTemperature===Number(range))
    return true;
return false
}
function getResults(){
    const inputTemperature =
document.getElementById('inputTemperature').value
    const dataStore = readStore()
    const responseData = [];
    dataStore.forEach(dataRow=>{
        const range = dataRow['Optimum growth temperature
(℃)'];
        if(isTemperatureInRange(inputTemperature,range))
            responseData.push(dataRow);
    })
    let response =""
    if(responseData.length===0)
    {
        response="<center>No data for the selected temperature!
Please input a diferent value.</center>"
    }
    else{
        response="<table><tr>"
    for(let key in responseData[0])
    {
        response += "<th>"+key+"</th>"
    }
```

```
    response+="</tr>"
    responseData.forEach((responseDataRow,index)=>{
        response+="<tr>"
        for(let key in responseDataRow)
        {
            if(key==='Website')
                response += "<td><a target='_blank'
href='responseDataRow[key]'>"+responseDataRow[key]+"</a></td>"
            else if(key==='S.No')
                    response += "<td>"+(index+1)+"</td>"
                else
                    response +=
"<td>"+responseDataRow[key]+"</td>"
        }
        response+="</tr>"
    });
    response+="</table>"
}
    document.getElementById('data-display').innerHTML=response;
    document.querySelector('body').addEventListener('click',
event => {
            if (event.target.tagName.toLowerCase() === 'a') {
                event.preventDefault();
                shell.openExternal(event.target.href);
            }
        });
}
```

All the previous code is for the different elements of the program. When the program starts, it initially reads the `index.html` file which then starts the flow of the program.

`Index.css` is the file that governs how the program will look and all the GUI elements are defined in this file. This is the file that is read as soon as GUI is invoked.

Next `index.js` file is started since this is the file that boots up JavaScript and loads up its components for further use by the program. This is basically from where the actual program starts. This is also the file which decides what frameworks and system resources would be required by the program.

Then, `renderer.js` file is run. This is the core of our program and is the file in which main logic of the program resides. This makes this file the most important one and the one in which any changes should be made with the utmost care, since any change can have impact on logic and flow of the program.

The database is however kept in a separate file to make the program much more modular. This will help in easy and frequent updating of the database. It is kept in a json (JavaScript object notation) file named `data_plot.json`. This file is called upon by the renderer.js file as and when needed by the program.

Data_plot.json


```
{
        {
            "S.No": "1",
            "Organism": "Pseudoalteromonas translucida",
            "Optimum growth temperature (℃)": "0-29",
            "Average Temperature": "14.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "1900",
            "Organism Category": "Bacteria"
```

```
        },
        {
            "S.No": "2",
            "Organism": "Methanothermobacter
thermautotrophicus",
            "Optimum growth temperature (℃)": "65-70",
            "Average Temperature": "67.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "5900",
            "Organism Category": "Bacteria"
        },
        {
            "S.No": "3",
            "Organism": "Brucella suis",
            "Optimum growth temperature (℃)": "37-40",
            "Average Temperature": "38.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "8900",
            "Organism Category": "Bacteria"
        },
        {
            "S.No": "4",
            "Organism": "Porphyromonas gingivalis",
            "Optimum growth temperature (℃)": "47-41",
            "Average Temperature": "44",
            "Specificity (KCat/KM) [1/mMs-1] ": "15000",
            "Organism Category": "Bacteria"
        },
        {
```

```
    "S.No": "5",
    "Organism": "Synechocystis sp. PCC 6803",
    "Optimum growth temperature (℃)": "32-38",
    "Average Temperature": "35",
    "Specificity (KCat/KM) [1/mMs-1] ": "17300",
    "Organism Category": "Bacteria"
},
{
    "S.No": "6",
    "Organism": "Persephonella marina",
    "Optimum growth temperature (℃)": "55-80",
    "Average Temperature": "67.5",
    "Specificity (KCat/KM) [1/mMs-1] ": "30000",
    "Organism Category": "Bacteria"
},
{
    "S.No": "7",
    "Organism": "Enterobacter hormaechei",
    "Optimum growth temperature (℃)": "35",
    "Average Temperature": "35",
    "Specificity (KCat/KM) [1/mMs-1] ": "56000",
    "Organism Category": "Bacteria"
},
{
    "S.No": "8",
    "Organism": "Nostoc commune",
    "Optimum growth temperature (℃)": "28",
```

```
    "Average Temperature": "28",
    "Specificity (KCat/KM) [1/mMs-1] ": "83000",
    "Organism Category": "Bacteria"
},
{
    "S.No": "9",
    "Organism": "Sulfurihydrogenibium sp. YO3AOP1",
    "Optimum growth temperature (℃)": "50-73",
    "Average Temperature": "61.5",
    "Specificity (KCat/KM) [1/mMs-1] ": "110000",
    "Organism Category": "Bacteria"
},
{
    "S.No": "10",
    "Organism": "Thermovibrio ammonificans",
    "Optimum growth temperature (℃)": "60-80",
    "Average Temperature": "70",
    "Specificity (KCat/KM) [1/mMs-1] ": "160000",
    "Organism Category": "Bacteria"
},
{
    "S.No": "11",
    "Organism": "Sulfurihydrogenibium azorense",
    "Optimum growth temperature (℃)": "80",
    "Average Temperature": "80",
    "Specificity (KCat/KM) [1/mMs-1] ": "350000",
    "Organism Category": "Bacteria"
```

```
},
{
    "S.No": "12",
    "Organism": "Methanosarcina thermophila",
    "Optimum growth temperature (℃)": "55-75",
    "Average Temperature": "65",
    "Specificity (KCat/KM) [1/mMs-1] ": "62900",
    "Organism Category": "Bacteria"
},
{
    "S.No": "13",
    "Organism": "Columba livia",
    "Optimum growth temperature (℃)": "45-50",
    "Average Temperature": "47.5",
    "Specificity (KCat/KM) [1/mMs-1] ": "110000",
    "Organism Category": "Animal"
},
{
    "S.No": "14",
    "Organism": "Danio rerio",
    "Optimum growth temperature (℃)": "26-28.5",
    "Average Temperature": "27.25",
    "Specificity (KCat/KM) [1/mMs-1] ": "130000",
    "Organism Category": "Animal"
},
{
    "S.No": "15",
```

```
            "Organism": "Homo sapiens",
            "Optimum growth temperature (℃)": "20-26",
            "Average Temperature": "23",
            "Specificity (KCat/KM) [1/mMs-1] ": "150000",
            "Organism Category": "Animal"
        },
        {
            "S.No": "16",
            "Organism": "Thalassiosira weissflogii",
            "Optimum growth temperature (℃)": "0",
            "Average Temperature": "0",
            "Specificity (KCat/KM) [1/mMs-1] ": "33000",
            "Organism Category": "Algae"
        },
        {
            "S.No": "17",
            "Organism": "Sordaria macrospora",
            "Optimum growth temperature (℃)": "15-36",
            "Average Temperature": "25.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "1210",
            "Organism Category": "Fungus"
        }
    ],
    "For database": [
        {
            "S.No": "1",
            "Organism": "Pseudoalteromonas translucida",
```

```
            "Optimum growth temperature (℃)": "0-29",
            "Average Temperature": "14.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "1900",
            "Organism Category": "Bacteria",
            "Temperature Category": "Psychrophilic",
            "Website": "https://www.uniprot.org/uniprot/Q3IJH5"
        },
        {
            "S.No": "2",
            "Organism": "Methanothermobacter
thermautotrophicus",
            "Optimum growth temperature (℃)": "65-70",
            "Average Temperature": "67.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "5900",
            "Organism Category": "Bacteria",
            "Temperature Category": "Thermophilic",
            "Website": "https://www.uniprot.org/uniprot/Q50565"
        },
        {
            "S.No": "3",
            "Organism": "Brucella suis",
            "Optimum growth temperature (℃)": "37-40",
            "Average Temperature": "38.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "8900",
            "Organism Category": "Bacteria",
            "Temperature Category": "Mesophilic",
            "Website": "https://www.uniprot.org/uniprot/A9WZ81"
```

```
},
{
    "S.No": "4",
    "Organism": "Porphyromonas gingivalis",
    "Optimum growth temperature (℃)": "47-41",
    "Average Temperature": "44",
    "Specificity (KCat/KM) [1/mMs-1] ": "15000",
    "Organism Category": "Bacteria",
    "Temperature Category": "Mesophilic",
    "Website": "https://www.uniprot.org/uniprot/T2NA71"
},
{
    "S.No": "5",
    "Organism": "Synechocystis sp. PCC 6803",
    "Optimum growth temperature (℃)": "32-38",
    "Average Temperature": "35",
    "Specificity (KCat/KM) [1/mMs-1] ": "17300",
    "Organism Category": "Bacteria",
    "Temperature Category": "Mesophilic",
    "Website": "https://www.uniprot.org/uniprot/Q54735"
},
{
    "S.No": "6",
    "Organism": "Persephonella marina",
    "Optimum growth temperature (℃)": "55-80",
    "Average Temperature": "67.5",
    "Specificity (KCat/KM) [1/mMs-1] ": "30000",
```

```
            "Organism Category": "Bacteria",
            "Temperature Category": "Thermophilic",
            "Website": "https://www.uniprot.org/uniprot/C0QRB5"
        },
        {
            "S.No": "7",
            "Organism": "Enterobacter hormaechei",
            "Optimum growth temperature (℃)": "35",
            "Average Temperature": "35",
            "Specificity (KCat/KM) [1/mMs-1] ": "56000",
            "Organism Category": "Bacteria",
            "Temperature Category": "Mesophilic",
            "Website":
"https://www.uniprot.org/uniprot/A0A145Y7J2"
        },
        {
            "S.No": "8",
            "Organism": "Nostoc commune",
            "Optimum growth temperature (℃)": "28",
            "Average Temperature": "28",
            "Specificity (KCat/KM) [1/mMs-1] ": "83000",
            "Organism Category": "Bacteria",
            "Temperature Category": "Mesophilic",
            "Website": "https://www.uniprot.org/uniprot/P94170"
        },
        {
            "S.No": "9",
```

```
        "Organism": "Sulfurihydrogenibium sp. YO3AOP1",
        "Optimum growth temperature (℃)": "50-73",
        "Average Temperature": "61.5",
        "Specificity (KCat/KM) [1/mMs-1] ": "110000",
        "Organism Category": "Bacteria",
        "Temperature Category": "Thermophilic",
        "Website": "https://www.uniprot.org/uniprot/B2V8E3"
    },
    {
        "S.No": "10",
        "Organism": "Thermovibrio ammonificans",
        "Optimum growth temperature (℃)": "60-80",
        "Average Temperature": "70",
        "Specificity (KCat/KM) [1/mMs-1] ": "160000",
        "Organism Category": "Bacteria",
        "Temperature Category": "Thermophilic",
        "Website": "https://www.uniprot.org/uniprot/E8T502"
    },
    {
        "S.No": "11",
        "Organism": "Sulfurihydrogenibium azorense",
        "Optimum growth temperature (℃)": "80",
        "Average Temperature": "80",
        "Specificity (KCat/KM) [1/mMs-1] ": "350000",
        "Organism Category": "Bacteria",
        "Temperature Category": "Thermophilic",
        "Website": "https://www.uniprot.org/uniprot/C1DTU5"
```

```
        },
        {
            "S.No": "12",
            "Organism": "Methanosarcina thermophila",
            "Optimum growth temperature (℃)": "55-75",
            "Average Temperature": "65",
            "Specificity (KCat/KM) [1/mMs-1] ": "62900",
            "Organism Category": "Bacteria",
            "Temperature Category": "Thermophilic",
            "Website": "https://www.uniprot.org/uniprot/P40881"
        },
        {
            "S.No": "13",
            "Organism": "Columba livia",
            "Optimum growth temperature (℃)": "45-50",
            "Average Temperature": "47.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "110000",
            "Organism Category": "Animal",
            "Temperature Category": "Mesophilic",
            "Website":
"https://www.uniprot.org/uniprot/A0A2I0MWT1"
        },
        {
            "S.No": "14",
            "Organism": "Danio rerio",
            "Optimum growth temperature (℃)": "26-28.5",
            "Average Temperature": "27.25",
```

```
        "Specificity (KCat/KM) [1/mMs-1] ": "130000",
        "Organism Category": "Animal",
        "Temperature Category": "Mesophilic",
        "Website": "https://www.uniprot.org/uniprot/F1R454"
    },
    {
        "S.No": "15",
        "Organism": "Homo sapiens",
        "Optimum growth temperature (℃)": "20-26",
        "Average Temperature": "23",
        "Specificity (KCat/KM) [1/mMs-1] ": "150000",
        "Organism Category": "Animal",
        "Temperature Category": "Mesophilic",
        "Website": "https://www.uniprot.org/uniprot/P00915"
    },
    {
        "S.No": "16",
        "Organism": "Thalassiosira weissflogii",
        "Optimum growth temperature (℃)": "0",
        "Average Temperature": "0",
        "Specificity (KCat/KM) [1/mMs-1] ": "33000",
        "Organism Category": "Algae",
        "Temperature Category": "Psychrophilic",
        "Website": "https://www.uniprot.org/uniprot/Q5U9J1"
    },
    {
        "S.No": "17",
```

```
            "Organism": "Sordaria macrospora",
            "Optimum growth temperature (℃)": "15-36",
            "Average Temperature": "25.5",
            "Specificity (KCat/KM) [1/mMs-1] ": "1210",
            "Organism Category": "Fungus",
            "Temperature Category": "Mesophilic",
            "Website": "https://www.uniprot.org/uniprot/C1L336"
        }
    ]
}
```


Pre-requisites for running the software

If running the software from the executable file for the desired operating system, there is no as such pre-requisites, however if running it from console in development mode, following are the requirements.

- A 32bit or a 64bit supported computer with either Windows / Macintosh / Linux based operating system installed.
- Administrator privilege for installation of dependencies.
- Node.js installed on the computer.
- Git installed on the computer.
- Electron (a node,js framework) installed.

After all of this installed, start terminal or command prompt in the folder where these files are there and run the following commands.

- `npm start` -> This command is used to start the program locally. This is best used when making changes to the program and testing it.
- `npm run make` -> This is used to compile the program and make an executable file of it. This command will make an operating system and architecture specific program. For example, if someone running a 64 bit version of Ubuntu (Linux based operating system) runs this command, it would get an executable file which could be deployed in 64bit instances of Ubuntu independent of the computer.
- `Npm run build` -> This command can be used to create an executable for any operating system from any computer. For example, from Ubuntu 64bit installation, the executable can be made even for 64bit installation of Windows. However, to run this there is a slight change to be made in the `package.json` file. In the build area we have to keep the following variables –
    - `-w` for windows
    - `-m` for macintosh
    - `-l` for linux based operating systems.

Global CO_2 Emission data

Year	C02 emitted in Billion tonnes
1750	0.009350528
1751	0.009350528
1752	0.009354192
1753	0.009354192
1754	0.009357856
1755	0.00936152
1756	0.010006384
1757	0.010010048
1758	0.010013712
1759	0.010017376
1760	0.010017376
1761	0.01097368
1762	0.010977344
1763	0.010981008
1764	0.010984672
1765	0.010988336
1766	0.012259744
1767	0.012263408
1768	0.012267072
1769	0.012270736
1770	0.0122744
1771	0.01361176
1772	0.013615424
1773	0.013619088
1774	0.013622752
1775	0.013626416
1776	0.015037056
1777	0.01504072
1778	0.015044384
1779	0.015048048
1780	0.015055376
1781	0.016843408
1782	0.016847072
1783	0.0168544
1784	0.016858064
1785	0.016869056
1786	0.019151728
1787	0.019159056
1788	0.01916272
1789	0.019170048
1790	0.019177376
1791	0.021419744
1792	0.021896064
1793	0.021914384
1794	0.021881408
1795	0.0218924
1796	0.022951296
1797	0.024094464
1798	0.025094736
1799	0.026428432
1800	0.028091888
1801	0.027959984
1802	0.036782896
1803	0.031488416
1804	0.034309696
1805	0.033419344
1806	0.03504616
1807	0.036874496
1808	0.03506448
1809	0.035090128
1810	0.037380128
1811	0.039582192
1812	0.041007488
1813	0.04122
1814	0.042128672
1815	0.043488016
1816	0.047664976
1817	0.049431024
1818	0.049643536
1819	0.049947648
1820	0.050687776
1821	0.051435232
1822	0.053465088
1823	0.056550176
1824	0.058525072
1825	0.060756448
1826	0.061419632
1827	0.06591536
1828	0.066637168
1829	0.066395344
1830	0.089123136
1831	0.08452848

1832	0.085111056
1833	0.086807488
1834	0.0884856
1835	0.09044584
1836	0.104775744
1837	0.104691472
1838	0.108044032
1839	0.111612768
1840	0.118929776
1841	0.122619424
1842	0.12950408
1843	0.132852976
1844	0.141423072
1845	0.155210704
1846	0.157793824
1847	0.172402192
1848	0.173816496
1849	0.18351144
1850	0.196896032
1851	0.198804976
1852	0.207550944
1853	0.217209248
1854	0.255138976
1855	0.260165984
1856	0.27729152
1857	0.279889296
1858	0.284172032
1859	0.301312853
1860	0.330642487

1861	0.347735471
1862	0.354057716
1863	0.377860218
1864	0.407091374
1865	0.43230805
1866	0.445970894
1867	0.477814912
1868	0.490752496
1869	0.52113072
1870	0.532536752
1871	0.56614296
1872	0.626371792
1873	0.66541904
1874	0.622997248
1875	0.675799152
1876	0.68555272
1877	0.699098528
1878	0.704382543
1879	0.754870251
1880	0.853707835
1881	0.88240807
1882	0.931925349
1883	0.991036392
1884	1.002178535
1885	1.009675266
1886	1.025479625
1887	1.076762379
1888	1.192276903
1889	1.191805414

1890	1.298464659
1891	1.358880878
1892	1.370084623
1893	1.353676428
1894	1.400866167
1895	1.485283456
1896	1.533713302
1897	1.606312994
1898	1.694279862
1899	1.85090877
1900	1.953614215
1901	2.018420559
1902	2.069946153
1903	2.258613447
1904	2.282759337
1905	2.431144771
1906	2.554215361
1907	2.88783285
1908	2.779287824
1909	2.89065356
1910	3.032155998
1911	3.087264469
1912	3.233403021
1913	3.498265954
1914	3.173995825
1915	3.130388638
1916	3.378662488
1917	3.53355565
1918	3.483434724

1919	3.019921454
1920	3.513394521
1921	3.083122765
1922	3.234231434
1923	3.670605469
1924	3.682986363
1925	3.709803778
1926	3.642199735
1927	3.9783737
1928	3.959653143
1929	4.252189759
1930	3.919398903
1931	3.502982456
1932	3.157845139
1933	3.324971535
1934	3.616315386
1935	3.794456537
1936	4.168554366
1937	4.45665482
1938	4.193545888
1939	4.435792518
1940	4.847084264
1941	4.955476469
1942	4.935659098
1943	5.018864796
1944	5.101039826
1945	4.24082889
1946	4.629673019
1947	5.125894209

1948	5.396503436
1949	5.239276402
1950	5.998350483
1951	6.373784977
1952	6.460163088
1953	6.641784991
1954	6.784943963
1955	7.437038185
1956	7.917910418
1957	8.179339626
1958	8.412113102
1959	8.848508658
1960	9.334894235
1961	9.356003209
1962	9.687509136
1963	10.23603084
1964	10.769609
1965	11.26928938
1966	11.79383789
1967	12.17149656
1968	12.83599584
1969	13.69184209
1970	14.82686256
1971	15.42583127
1972	16.14260518
1973	17.00071304
1974	16.92587153
1975	16.90297254
1976	17.79976025

1977	18.28793609
1978	18.95883194
1979	19.46423814
1980	19.36945179
1981	18.84136658
1982	18.70096497
1983	18.87627456
1984	19.42623405
1985	20.11682422
1986	20.4016947
1987	21.06270396
1988	21.86597075
1989	22.19336178
1990	22.69761192
1991	23.16952933
1992	22.44495133
1993	22.68260321
1994	22.84378772
1995	23.33215907
1996	24.05084364
1997	24.19114815
1998	24.11211881
1999	24.43105076
2000	25.11904231
2001	25.33220252
2002	25.91118628
2003	27.17618398
2004	28.47045125
2005	29.41088934

2006	30.37455399
2007	31.2938623
2008	31.9460339
2009	31.46420043
2010	33.13191107

2011	34.20958257
2012	34.76000814
2013	34.98726355
2014	35.24486811
2015	35.20944656

2016	35.22041242
2017	35.69634889
2018	36.41971171
2019	36.44138758

REFERENCES

1. Yong, J. K. J., Stevens, G. W., Caruso, F. & Kentish, S. E. The use of carbonic anhydrase to accelerate carbon dioxide capture processes: Carbonic anhydrase to accelerate carbon dioxide capture processes. *J. Chem. Technol. Biotechnol.* **90**, 3–10 (2015).

2. Carbon Dioxide Capture from Flue Gases Using a Cross-Flow Membrane Contactor and the Ionic Liquid 1-Ethyl-3-methylimidazolium Ethylsulfate | Industrial & Engineering Chemistry Research. https://pubs.acs.org/doi/abs/10.1021/ie1014266.

3. Olivier, J. G. J., Peters, J. a. H. W. & Janssens-Maenhout, G. Trends in global CO2 emissions. 2012 Report. (2012) doi:10.2788/33777.

4. CO2 emissions. *Our World in Data* https://ourworldindata.org/co2-emissions.

5. Sharma, A. & Bhattacharya, A. Enhanced biomimetic sequestration of CO2 into CaCO3 using purified carbonic anhydrase from indigenous bacterial strains. *Journal of Molecular Catalysis B: Enzymatic* **67**, 122–128 (2010).

6. Carbon capture and storage potential in coal-fired plant in Malaysia—A review - ScienceDirect. https://www.sciencedirect.com/science/article/abs/pii/S1364032110001504.

7. Change, I. P. on C. *Climate Change 2013: The Physical Science Basis: Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. (Cambridge University Press, 2014).

8. Bachu, S. Sequestration of CO2 in geological media: criteria and approach for site selection in response to climate change. *Energy Conversion and Management* **41**, 953–970 (2000).

9. Investigating the Application of Enzyme Carbonic Anhydrase for CO2 Sequestration Purposes | Industrial & Engineering Chemistry Research. https://pubs.acs.org/doi/abs/10.1021/ie060287u.

10. da Costa Ores, J., Sala, L., Cerveira, G. P. & Kalil, S. J. Purification of carbonic anhydrase from bovine erythrocytes and its application in the enzymic capture of carbon dioxide. *Chemosphere* **88**, 255–259 (2012).

11. Solubilization and concentration of carbon dioxide: Novel spray reactors with immobilized carbonic anhydrase - Bhattacharya - 2004 - Biotechnology and Bioengineering - Wiley Online Library. https://onlinelibrary.wiley.com/doi/abs/10.1002/bit.20042.

12. Aggarwal, M., Boone, C. D., Kondeti, B. & McKenna, R. Structural annotation of human carbonic anhydrases. *Journal of Enzyme Inhibition and Medicinal Chemistry* **28**, 267–277 (2013).

13. Biomimetic CO2 Sequestration: 1. Immobilization of Carbonic Anhydrase within Polyurethane Foam | Energy & Fuels. https://pubs.acs.org/doi/abs/10.1021/ef9005725.

14. Smith, K. S., Jakubzick, C., Whittam, T. S. & Ferry, J. G. Carbonic anhydrase is an ancient enzyme widespread in prokaryotes. *PNAS* **96**, 15184–15189 (1999).

15. Recent Advancements in Carbonic Anhydrase–Driven Processes for CO2 Sequestration: Minireview: Critical Reviews in Environmental Science and Technology: Vol 42, No 14. https://www.tandfonline.com/doi/abs/10.1080/10643389.2011.556884.

16. Hewett-Emmett, D. & Tashian, R. E. Functional diversity, conservation, and convergence in the evolution of the alpha-, beta-, and gamma-carbonic anhydrase gene families. *Mol Phylogenet Evol* **5**, 50–77 (1996).

17. Lindskog, S. Structure and mechanism of carbonic anhydrase. *Pharmacol Ther* **74**, 1–20 (1997).

18. Tripp, B. C., Smith, K. & Ferry, J. G. Carbonic Anhydrase: New Insights for an Ancient Enzyme. *Journal of Biological Chemistry* **276**, 48615–48618 (2001).

19. Iverson, T. M., Alber, B. E., Kisker, C., Ferry, J. G. & Rees, D. C. A closer look at the active site of gamma-class carbonic anhydrases: high-resolution

crystallographic studies of the carbonic anhydrase from Methanosarcina thermophila. *Biochemistry* **39**, 9222–9231 (2000).

20. PDB101: Molecule of the Month: Carbonic Anhydrase. *RCSB: PDB-101* http://pdb101.rcsb.org/motm/49.

21. A biological function for cadmium in marine diatoms | PNAS. https://www.pnas.org/content/97/9/4627.short.

22. Structure and metal exchange in the cadmium carbonic anhydrase of marine diatoms | Nature. https://www.nature.com/articles/nature06636.

23. Lapointe, M., Mackenzie, T. D. B. & Morse, D. An external delta-carbonic anhydrase in a free-living marine dinoflagellate may circumvent diffusion-limited carbon acquisition. *Plant Physiol* **147**, 1427–1436 (2008).

24. Berg, J. M., Tymoczko, J. L. & Stryer, L. *Biochemistry*. (W.H. Freeman and Company, 2012).

25. Carrillo, N., Ceccarelli, E. A. & Roveri, O. A. Usefulness of Kinetic Enzyme Parameters in Biotechnological Practice. *Biotechnology and Genetic Engineering Reviews* **27**, 367–382 (2010).

26. 18% Highly Polluting Industries Not Complying with Emissions Standards: Govt. https://thewire.in/environment/polluting-industries-emissions-standards.

27. Emissions by sector. *Our World in Data* https://ourworldindata.org/emissions-by-sector.

28. What is a Circular Economy? | Ellen MacArthur Foundation. https://www.ellenmacarthurfoundation.org/circular-economy/concept.

29. Weetman, C. *English: A simple diagram to contrast the 'take, make, waste' linear approach to the circular economy*. (2016).

30. Huang, C.-H. & Tan, C.-S. A Review: CO2 Utilization. *Aerosol Air Qual. Res.* **14**, 480–499 (2014).

31. File:The Circular Economy concept.png. *Wikipedia*.

32. Bhardwaj, R. *et al.* Integrated catalytic insights into methanol production: Sustainable framework for CO2 conversion. *Journal of Environmental Management* **289**, 112468 (2021).

33. Strong, P. J. *et al.* The Opportunity for High-Performance Biomaterials from Methane. *Microorganisms* **4**, (2016).

34. Carbon Capture. *Center for Climate and Energy Solutions* https://www.c2es.org/content/carbon-capture/ (2017).

35. Efficient reduction of CO2 using a novel carbonic anhydrase producing Corynebacterium flavescens. http://eeer.org/journal/view.php?number=1182.

36. Novel disulfide engineering in human carbonic anhydrase II using the PAIRWISE side-chain geometry database - Burton - 2000 - Protein Science - Wiley Online Library. https://onlinelibrary.wiley.com/doi/abs/10.1110/ps.9.4.776.

37. Barbero, R. *et al.* Engineered yeast for enhanced CO2 mineralization. *Energy Environ. Sci.* **6**, 660–674 (2013).

38. Engineered Escherichia coli with Periplasmic Carbonic Anhydrase as a Biocatalyst for CO2 Sequestration. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3811487/.

39. Chemistry for Biologists. *RSB* www.rsb.org.uk/education/teaching-resources/secondary-schools/chemistry-for-biologists.

40. Imtaiyaz Hassan, Md., Shajee, B., Waheed, A., Ahmad, F. & Sly, W. S. Structure, function and applications of carbonic anhydrase isozymes. *Bioorganic & Medicinal Chemistry* **21**, 1570–1582 (2013).

41. Unknown. About Node.js, and why you should add Node.js to your skill set? http://blog.training.com/2016/09/about-nodejs-and-why-you-should-add.html.

42. Joyent & Node. https://groups.google.com/g/nodejs/c/lWo0MbHZ6Tc?pli=1.

43. What You Need To Know About Node.js. *ReadWrite* https://readwrite.com/2013/11/07/what-you-need-to-know-about-nodejs/ (2013).

44. Node.js. About. *Node.js* https://nodejs.org/en/about/.

45. Olsen, S. E., Tangstad, M. & Lindstad, T. *Production of manganese ferroalloys*. (Tapir Akademisk Forlag, 2007).

46. 6.9B: Classification of Microorganisms by Growth Temperature. *Biology LibreTexts* https://bio.libretexts.org/Bookshelves/Microbiology/Book%3A_Microbiology_(Boundless)/6%3A_Culturing_Microorganisms/6.9%3A_Temperature_and_Microbial_Growth/6.9B%3A_Classification_of_Microorganisms_by_Growth_Temperature (2017).

47. PSHAa2765 - Carbonic anhydrase - Pseudoalteromonas translucida (strain TAC 125) - PSHAa2765 gene & protein. https://www.uniprot.org/uniprot/Q3IJH5.

48. MTH_1582 - Carbonic anhydrase - Methanothermobacter thermautotrophicus (strain ATCC 29096 / DSM 1053 / JCM 10044 / NBRC 100330 / Delta H) - MTH_1582 gene & protein. https://www.uniprot.org/uniprot/Q50565.

49. BSUIS_B0780 - Carbonic anhydrase - Brucella suis (strain ATCC 23445 / NCTC 10510) - BSUIS_B0780 gene & protein. https://www.uniprot.org/uniprot/A9WZ81.

50. cah - Carbonate dehydratase - Porphyromonas gingivalis JCVI SC001 - cah gene & protein. https://www.uniprot.org/uniprot/T2NA71.

51. ccaA - Carbonic anhydrase - Synechocystis sp. (strain PCC 6803 / Kazusa) - ccaA gene & protein. https://www.uniprot.org/uniprot/Q54735.

52. PERMA_1443 - Carbonic anhydrase - Persephonella marina (strain DSM 14350 / EX-H1) - PERMA_1443 gene & protein. https://www.uniprot.org/uniprot/C0QRB5.

53. cah - Carbonic anhydrase - Enterobacter hormaechei - cah gene & protein. https://www.uniprot.org/uniprot/A0A145Y7J2.

54. ecaA - Carbonic anhydrase precursor - Nostoc sp. (strain PCC 7120 / SAG 25.82 / UTEX 2576) - ecaA gene & protein. https://www.uniprot.org/uniprot/P94170.

55. SYO3AOP1_0578 - Carbonic anhydrase precursor - Sulfurihydrogenibium sp. (strain YO3AOP1) - SYO3AOP1_0578 gene & protein. https://www.uniprot.org/uniprot/B2V8E3.

56. Theam_1576 - Carbonic anhydrase precursor - Thermovibrio ammonificans (strain DSM 15698 / JCM 12110 / HB-1) - Theam_1576 gene & protein. https://www.uniprot.org/uniprot/E8T502.

57. SULAZ_0541 - Carbonic anhydrase - Sulfurihydrogenibium azorense (strain Az-Fu1 / DSM 15241 / OCM 825) - SULAZ_0541 gene & protein. https://www.uniprot.org/uniprot/C1DTU5.

58. Carbonic anhydrase precursor - Methanosarcina thermophila. https://www.uniprot.org/uniprot/P40881.

59. A306_00001363 - Carbonic anhydrase - Columba livia (Rock dove) - A306_00001363 gene & protein. https://www.uniprot.org/uniprot/A0A2I0MWT1.

60. ca2 - Carbonic anhydrase - Danio rerio (Zebrafish) - ca2 gene & protein. https://www.uniprot.org/uniprot/F1R454.

61. CA1 - Carbonic anhydrase 1 - Homo sapiens (Human) - CA1 gene & protein. https://www.uniprot.org/uniprot/P00915.

62. ca1 - Delta carbonic anhydrase - Thalassiosira weissflogii (Marine diatom) - ca1 gene & protein. https://www.uniprot.org/uniprot/Q5U9J1.

63. cas2 - Carbonic anhydrase - Sordaria macrospora - cas2 gene & protein. https://www.uniprot.org/uniprot/C1L336.