

# Informe: Proyecto de IA-Simulación

Integrantes:

Jose Carlos Pendás Rodríguez C-411

Max Bengochea Moré C-411

Maykol Luis Martínez Rodríguez C-412

April 28, 2024

# 1 Introducción al problema, simulación y agentes:

El sistema en el que se centra nuestra simulación tiene como objetivo modelar el funcionamiento de un conjunto de empresas y sus interacciones con el mercado. Es un sistema episódico y no determinista. Como está más allá de nuestra capacidad asignar un comportamiento inteligente a todas las entidades involucradas en una economía nacional, el tipo de agente creado se basa en compañías mientras que los consumidores finales y otras compañías no agentes son simuladas, en pocas palabras, definimos el comportamiento de un cierto grupo de compañías y su relación con el resto del mercado global. Estas compañías o empresas, son capaces de comprar, vender, construir nuevas industrias para la producción y usar esas industrias para transformar un producto en otro. También tenemos el mercado, que es el encargado de regular la compra y venta que realizan las industrias, la venta se realiza lógicamente en función de los precios mientras que la compra depende del azar. En el mercado también existe una empresa fantasma que representa a otras empresas no agentes, así como a los consumidores finales de los productos, esta vende y compra una cierta cantidad de productos en cada iteración y varía sus precios en función de la oferta y la demanda. En la simulación también existen eventos que se desencadenan aleatoriamente como una superinflación que varía los precios de los productos utilizando datos históricos de la inflación en Venezuela y Brasil. Cada empresa solo puede realizar una de las cuatro acciones antes mencionada en la simulación, así mismo la cantidad de productos que puede almacenar y usar como materia prima en la elaboración de otros productos depende de la cantidad de industrias que posea.

## 2 Búsqueda:

En cuanto a la IA de nuestros agentes, además del clásico algoritmo random y greedy, se implementó un algoritmo Uniform Cost Search, dado que el costo de cada acción es consumir un turno de la empresa, el resultado final será se comportará de manera similar a un BFS, el propósito de esto es encontrar el camino más corto para lograr un cierto crecimiento para la empresa (que será medido en función de sus acciones y su capital), en otras palabras, un nodo se considerará final si la empresa ha crecido en un cierto porcentaje en relación al nodo actual, definir el valor de este porcentaje tendrá naturalmente un impacto profundo en la estrategia, un porcentaje muy bajo no lo hace muy

diferente de una estrategia greedy, mientras que un porcentaje mayor permite hacer planificaciones a largo plazo, esto nos permite analizar el impacto de la planificación a largo plazo en la gestión de una industria. Además de las acciones mencionadas, las empresas son capaces de establecer convenios entre ellas, solo que aún no se ha configurado por completo la IA que le permita hacer esto, esperamos esté completa para la exposición.

### 3 Conocimiento:

Como se modela la toma de decisiones de las empresas, según el componente de conocimiento:

Cada empresa tiene un conjunto finito de decisiones que puede tomar, como en cada iteración de la simulación solo se puede tomar una, una de las soluciones que contemplamos fue definir un sistema experto basado en Lógica difusa, el cual funciona a partir de, dado ciertas condiciones (Antecedentes), se valora que tan importante es una acción específica, los valores de esta importancia serían Consecuentes en nuestro sistema. Contemplamos el uso de este sistema experto para modelar la toma de decisiones pues estos son buenos para resolver ambigüedades basado en conocimiento adquirido, este conocimiento es el que representamos en reglas, a partir de las cuales se dictamina cual acción es mas importante dadas ciertas circunstancias.

Como esta construido el sistema de control:

Cada variable de la simulación pudiera representar un universo discurso, sobre el cual se pudiera definir funciones de membresía, por ejemplo, digamos que la empresa tIron que se dedica a refinar hierro, tiene en sus almacenes una cantidad  $x$  de hierro refinado y quiere definir que tan beneficioso pudiera ser vender hierro en esta iteración de la simulación, podríamos representar el hierro de la empresa con un universo discurso que cubre los valores de 0 hasta su capacidad máxima de almacenamiento, sobre el cual se pudieran construir las siguientes funciones de membresía para una capacidad máxima de 100t de hierro refinado:

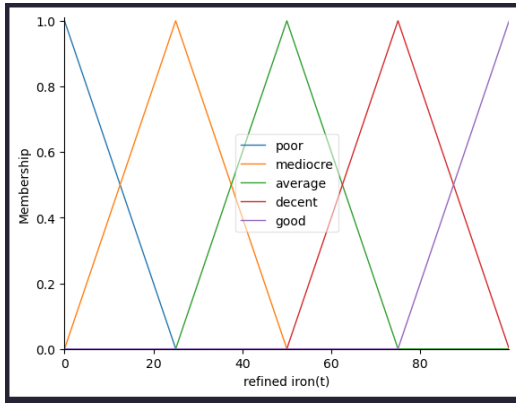


Figure 1: Función de membresía

Replicando esto en el universo discurso de la valoración de la acción, se tendrían estas funciones de membresía:

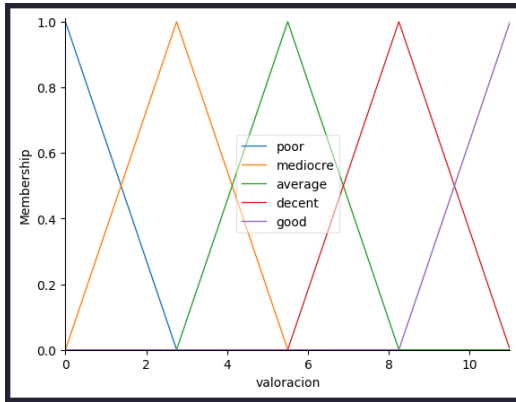


Figure 2: Función de membresía

Como ejemplo se pudiera definir las reglas:  $\text{rule1} = \text{refined iron}[\text{poor}] = \text{valoración}[\text{poor}]$   $\text{rule2} = \text{refined iron}[\text{mediocre}] = \text{valoración}[\text{mediocre}]$  ... y así sucesivamente para cada conjunto difuso, así se evitarían cuellos de botella en la producción, pues cuando los almacenes están llenos vender lo que tienes ganaría la máxima prioridad. Esto se puede complejizar tanto como se quiera, por ejemplo:

$\text{rule1} = \text{refined iron}[\text{good}] \text{ or } \text{iron.price}[\text{good}] = \text{valoración}[\text{good}]$   $\text{rule1} = \text{refined iron}[\text{decent}] \text{ or } \text{iron.price}[\text{decent}] = \text{valoración}[\text{decent}]$  etc. donde el

universo discurso del precio del hierro pudiera estar definido en una vecindad del valor medio del mismo, de esa forma si los almacenes están llenos igual tendrá toda la prioridad pues en la relación  $f(a)$  or  $g(b)$  en la lógica difusa resulta de calcular  $\text{Max}(f(a), g(b))$ , y en caso de que los precios estén muy por encima de la media tendrá bastante prioridad a pesar de no estar lleno el almacenamiento.

## 4 NLP:

Para llevar a cabo el procesamiento de lenguaje natural en nuestro proyecto nos apoyamos en la plataforma LM Studio. Por medio de esta, descargamos un modelo de lenguaje llamado Hermes 2 Mistral, con el cual interactuamos por medio de consultas. Para conectarnos con el modelo y así iniciar la comunicación, levantamos un servidor local a través de la plataforma en nuestra pc e ingresamos una url para iniciar la conexión. Como ventaja de este proceso figuran que no es necesario tener internet para realizar la comunicación de nuestro script de python con nuestro modelo y la plataforma cuenta con una interfaz bastante fácil de interactuar. Como desventaja figura la exigencia de nuestro modelo, pues se necesita un mínimo de 8 GB de RAM en nuestra PC, además de contar con una GPU de NVIDIA preferiblemente para garantizar una experiencia normal en cuanto a la velocidad, de lo contrario, deberíamos cargarlo usando completamente en nuestra CPU, lo cual resulta en un flujo de conexión bastante lento, demorando hasta 1 segundo o más por cada token generado en la respuesta.

### 4.1 PROBLEMAS QUE ATACAMOS CON NLP

Con nlp en nuestro proyecto nos procuramos solventar varias cuestiones. Al iniciar nuestra simulación, el usuario tiene acceso a las empresas de nuestro sistema económico, y puede escoger que estrategia sigue cada una. Para ello, luego de escoger la empresa a la cual desea designarle la estrategia seleccionada, el usuario ingresa una query en lenguaje natural donde define las reglas por las que se guiará nuestra empresa y el número de semanas (iteraciones de mercado) que adoptará este comportamiento. Luego, al terminar nuestra simulación, se crea un "informe de simulación" de todo lo sucedido durante la simulación. Este informe se lo pasamos a nuestro modelo, el cual procesa la información, y posteriormente se pueden realizar consultas en

lenguaje natural sobre los resultados obtenidos durante la simulacion (que empresa vendio mas productos, cuanto genero una empresa especifica en la n-esima semana, cuantos productos diferentes estuvieron disponibles en el mercado, etc...)

## 4.2 PROBLEMAS QUE ENCONTRAMOS EN EL PROCESO

Al tratarse de un modelo de lenguaje, al realizar varios tests (consultas hechas por nosotros mismos), nos percatamos que a veces las respuestas carecian de objetividad y la informacion en muchas ocasiones no era del todo certera. Por ejemplo: A veces podias preguntar que se vendio en el mercado, y podias recibir una respuesta tipo el mercado realizo..... (trataba el mercado como si fuera una empresa), o al preguntar por una accion especifica de una empresa, te daba TODO lo que realizo la empresa durante la simulacion. Otra cuestion era el tipico sms final de asistente virtual "Estoy aqui para ayudarte preguntame lo que quieras", el cual ya conocemos (muy similar al que nos manda chat-GPT)

## 4.3 COMO RESOLVIMOS ESTOS PROBLEMAS

Para resolver estas cuestiones primeramente realizamos un minucioso trabajo a la hora de crear nuestro prompt para designarle al modelo sus roles. Se necesito de gran especificacion en el mismo, para evitar que nuestro modelo se desvirtuara de las pregunta y garantizar objetividad. Con el trabajo de prompt, las respuestas ya se asemejaban a lo que queriamos saber, pero aun permanecia ese mensaje de "estoy aqui para ayudarte". Luego nos percatamos que al ajustar la respuesta en formato .json, las respuestas eran muchisimo mas concisas solidas y se asemejaba mas a lo que realmente queria saber el usuario (ya no divagaba tanto), ademas que desaparecia el sms final que tanto nos mortificaba. Esta opcion s bastante facil de configurar en la plataforma, pero tambien tenia sus cuestiones. Por ejemplo una conversacion con nuestro modelo con la respuesta configurada en .json, puede ser del tipo: -USER: "Cuantos mundiales tiene Italia" -RESPONSE: "Italia": "Tienes 4 mundiales" "que ganos": "en las fechas (fechas de los mundiales de talia)" Esto se ve un poco raro, pero necesario, puesto que ganamos en objetividad. Luego para que nuestra respuesta no sea asi, en el prompt le especificamos que esperamos un .json con los campos siguientes, le indicamos

que informacion almacenar en cada campo luego de procesarla y que tipo se informacion, ya sea un numero o simplemente texto. Esto nos obliga a mediante el uso de la biblioteca json de python, parsear la respuesta, pero es un precio bastante barato a pagar por lo que ganamos en objtividad y solidez a la hora de responder las consultas

## 5 Resultados:

Otra observación importante realizada es la situación a la que se ven expuestas las compañías al enfrentarse a un fenómeno de hiperinflación, la siguiente gráfica muestra el comportamiento de la inflación:

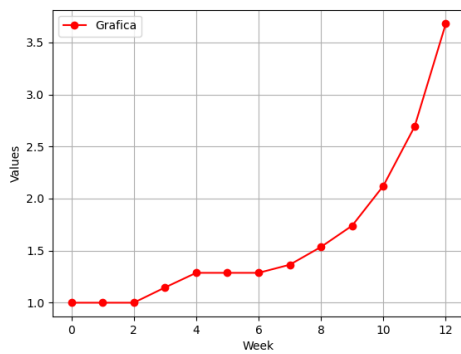


Figure 3: Función de membresía

Mientras que esta tabla nos muestra el descenso económico observado en una de las compañías durante el mismo período:

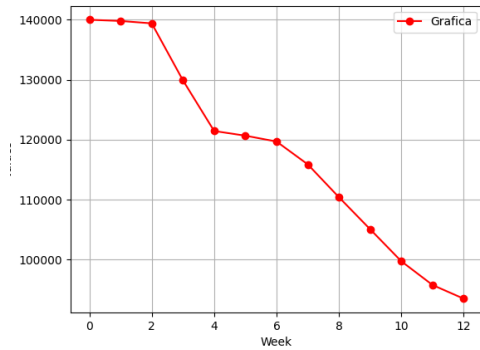


Figure 4: Función de membresía

Este resultado no es sorprendente teniendo en cuenta que los datos utilizados son referentes a registros de las mayores hiperinflaciones de la historia, dado esto se pone en evidencia en impacto de uno de los fenómenos económicos más dañinos de la actualidad.

## 6 Investigaciones futuras:

Además de la inflación existen otros fenómenos que afectan a la economía, algunos ambientales otros sociales, nuestro objetivo futuro es simular todos estos eventos y observar la reacción de las empresas a estos, permitiendo obtener datos sobre su posible impacto y quizás estrategias para minimizarlo.