Applied Data Science Project

“Generation of video reviews based on textual descriptions”

- Objective(s)
- Research question(s)
- Methods
- Experiments
- Conclusions

FLOWYGO

FLOWYGO

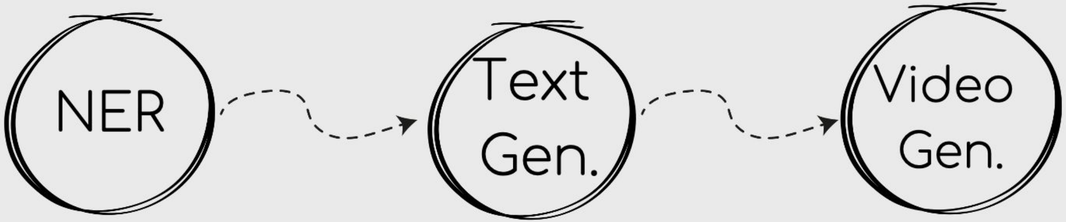
SUSTAINABLE DEVELOPMENT GOALS

12 RESPONSIBLE CONSUMPTION AND PRODUCTION

Project

(objectives)

- Goal: a system able to generate text and video reviews from a list of features of a given product
- The objective is to increase the chance of a better match between customer and products
- The system is divided in three main components:



SUSTAINABLE DEVELOPMENT GOALS

12 RESPONSIBLE CONSUMPTION AND PRODUCTION

Project

(research questions)

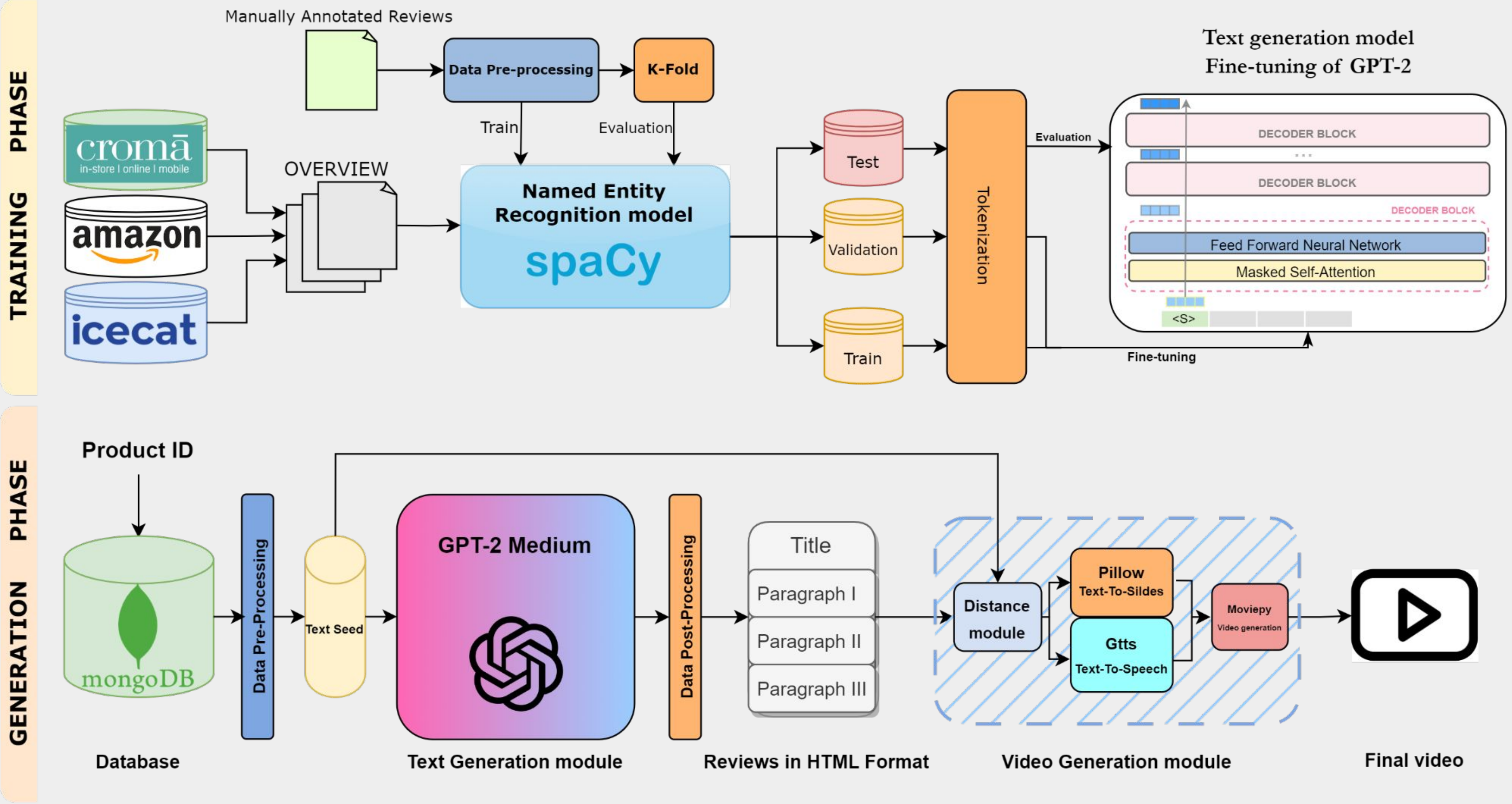
Main criticalities/research questions:

- How can we build a complete pipeline capable of automatically generating a video review starting from the specifics of a tech product?
- How can we efficiently synchronize the information contained in a textual review with the review itself?
- How can we exploit the scalability of the GPT-2 system considering the computational cost?

We will further show:

- Description of how NER and GPT-2 are applied
- Description of how the video generation works
- Evaluation metrics for NER and GPT-2 models

Methods


TRAINING PHASE
Manually Annotated Reviews
Data Pre-processing
K-Fold
Train
Evaluation
croma in-store | online | mobile
amazon
icecat
OVERVIEW
Named Entity Recognition model
spaCy
Test
Validation
Train
Tokenization
Evaluation
Fine-tuning
Text generation model
Fine-tuning of GPT-2
DECODER BLOCK
...
DECODER BLOCK
DECODER BOLCK
Feed Forward Neural Network
Masked Self-Attention
<S>
GENERATION PHASE
Product ID
mongoDB
Data Pre-Processing
Text Seed
GPT-2 Medium
Data Post-Processing
Title
Paragraph I
Paragraph II
Paragraph III
Distance module
Pillow
Text-To-Sildes
Gtts
Text-To-Speech
Moviepy
Video generation
Database
Text Generation module
Reviews in HTML Format
Video Generation module
Final video

spaCy

NER

Named Entity Recognizer

- We used the spaCy's Named Entity Recognizer (NER).
- We customized the NER to adapt it to our task.
- It has the capability of identifying the features and the name of a product inside a review of such a product.
- In our case it was used to generate the training dataset for the GPT-2 fine tuning.

Model for the text generation

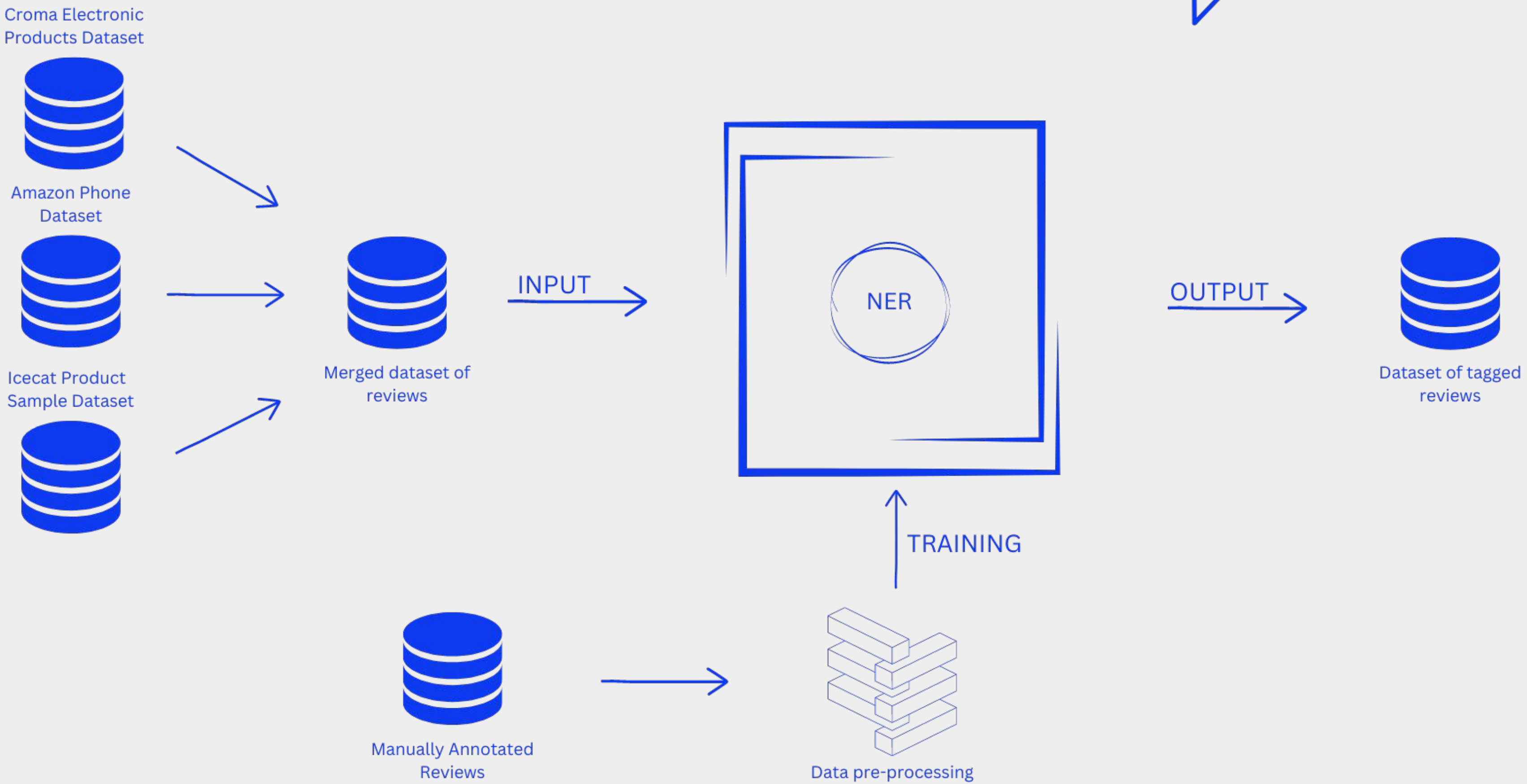
GPT-2 Medium
A larger model

- 345M Parameters
- 1024 Model Dimensionality

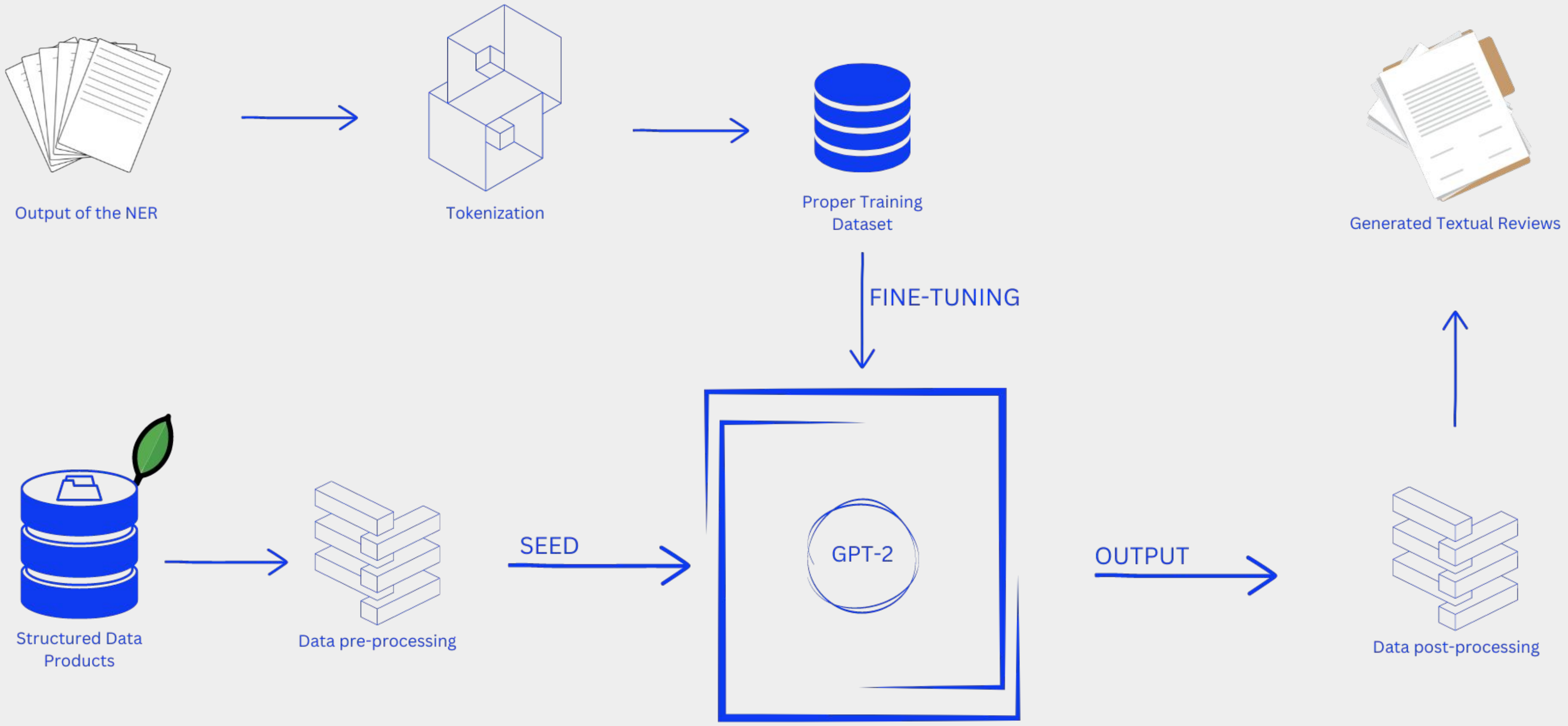
- For the textual reviews generation we used the GPT-2 medium.
- GPT-2 Medium is the 355M parameter version of GPT-2, a transformer-based language model created and released by OpenAI.
- It is pre-trained on a set of about 40GB called WebText.

FLOWYGO

What does the NER do?


Croma Electronic Products Dataset
Amazon Phone Dataset
Icecat Product Sample Dataset
Merged dataset of reviews
INPUT
NER
OUTPUT
Dataset of tagged reviews
TRAINING
Manually Annotated Reviews
Data pre-processing

GPT-2



Output of the NER

Tokenization

Proper Training Dataset

FINE-TUNING

Generated Textual Reviews

Structured Data Products

Data pre-processing

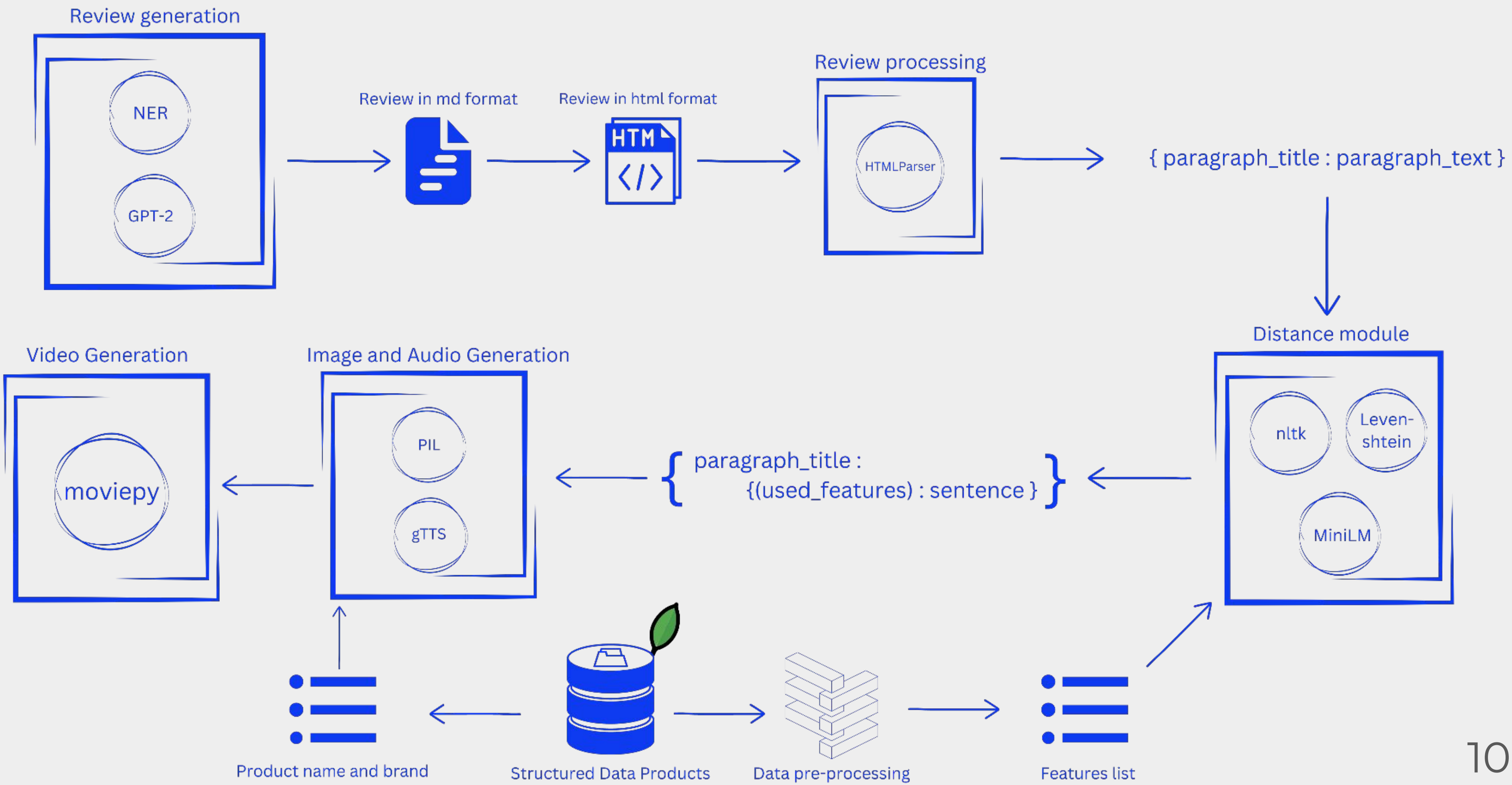
SEED

GPT-2

OUTPUT

Data post-processing

Video Generation



Levenshtein distance

- Minimum number of single character edits needed to transform a string into the other

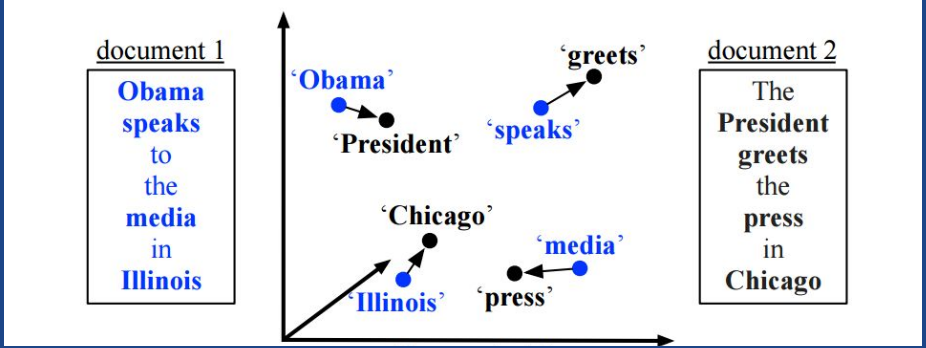
$$\text{hello} \xrightarrow{\text{deletion}} \text{hell}\cancel{\text{o}} \xrightarrow{\text{substitution}} \text{help}$$

$$\Longrightarrow \quad \text{lev(hellp, help)} = 2$$

- Pros:
  - classical string metric, directly applied on the sentences
- Cons:
  - doesn’t generalize well to sentences
- Implementation notes:
  - applied on all the permutations of words contained in windows of the sentences

Sentence embedding

- Sentences are mapped to real-valued vectors, then cosine similarity is computed



- Pros:
  - encapsulate the semantic of a sentence
- Cons:
  - extra step are introduced
- Implementation notes:
  - all-MiniLM-L6-v2 is used, which is distilled from BERT

Metrics

Output comparison

- We can identify two types of match:
  - perfect match
  - semantic match

- Comparing the two approaches:
  - sentence embedding sees approximately a loss in perfect matches of 7.9%
  - sentence embedding allows for semantic matches, e.g.:
    - slim □ flat screen
    - greater viewing area□ 65 inches
    - remote is easy to operate and the remote button is easy to press □ remote one touch access

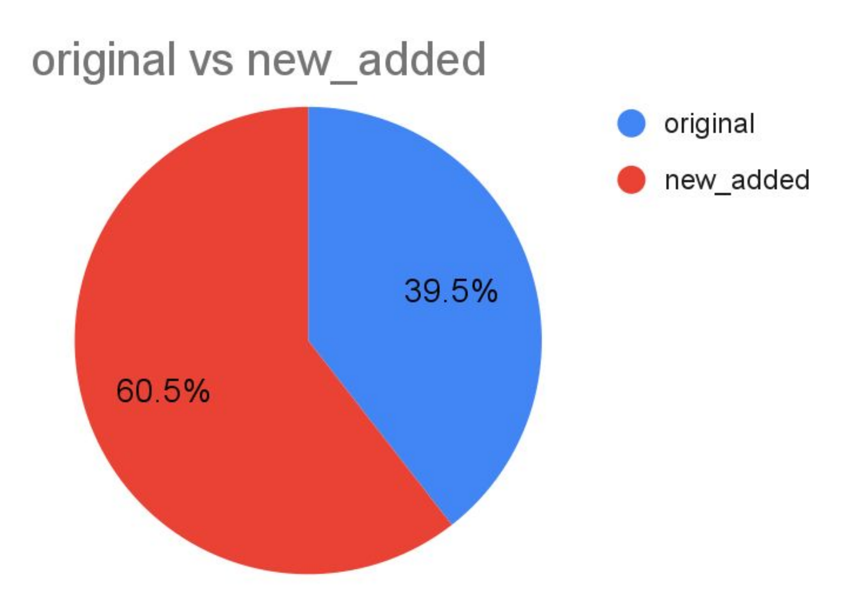
Experiments

NER

Dataset

Dataset details:

- 146 manually annotated reviews (triplicated the original dataset)
- 328 product name tags
- 1697 attribute name tags



NER

Evaluation

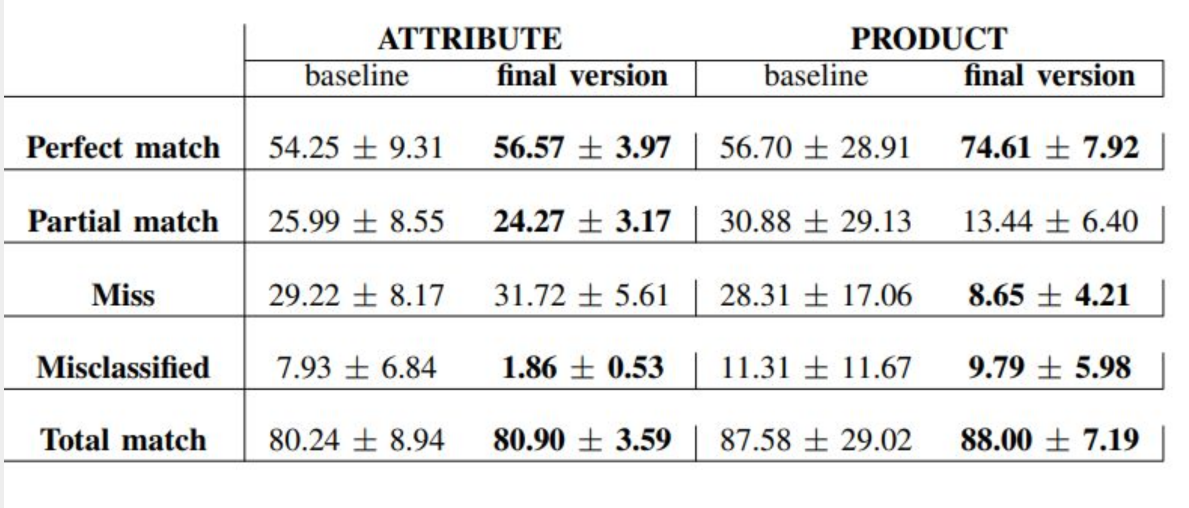
Evaluation method:

- K-Fold with 10 split on the manually annotated reviews

Metrics for evaluation :

- Perfect match
- Partial match
- Missed
- Misclassified
- Total match

NER Performance : baseline vs final model



	ATTRIBUTE		**PRODUCT**	
	baseline	**final version**	baseline	**final version**
Perfect match	54.25 ± 9.31	**56.57 ± 3.97**	56.70 ± 28.91	**74.61 ± 7.92**
Partial match	25.99 ± 8.55	**24.27 ± 3.17**	30.88 ± 29.13	13.44 ± 6.40
Miss	29.22 ± 8.17	31.72 ± 5.61	28.31 ± 17.06	**8.65 ± 4.21**
Misclassified	7.93 ± 6.84	**1.86 ± 0.53**	11.31 ± 11.67	**9.79 ± 5.98**
Total match	80.24 ± 8.94	**80.90 ± 3.59**	87.58 ± 29.02	**88.00 ± 7.19**

Main focus: Overall increasing performance

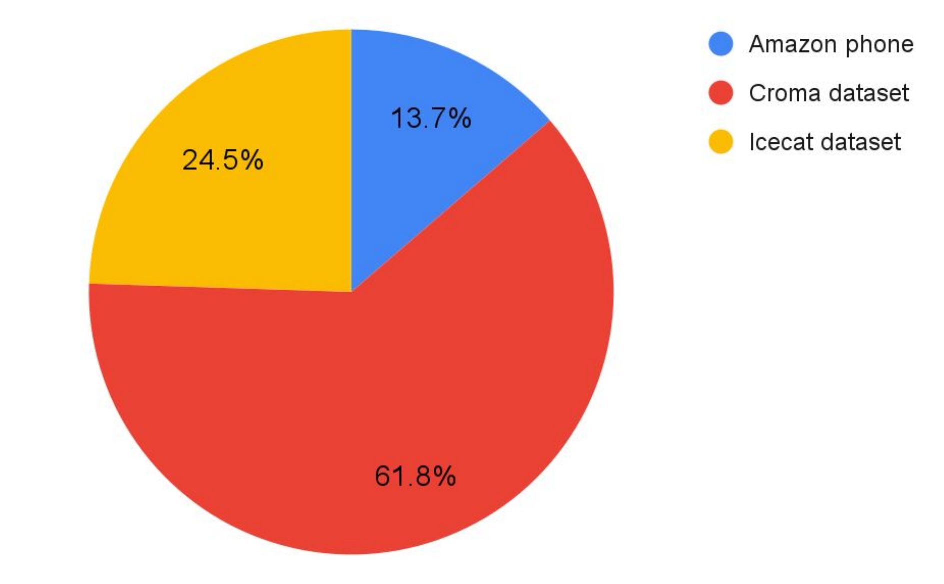
- In attribute performance, drastically decreased the error of misclassifying an attribute with product
- In product performance, we increased the performance in perfect match of the product name

GPT-2

Dataset

Dataset size:

- 24792 records: each one normalized into (product name, description)
- 90-8-2 train valid test



GPT-2

Evaluation metrics

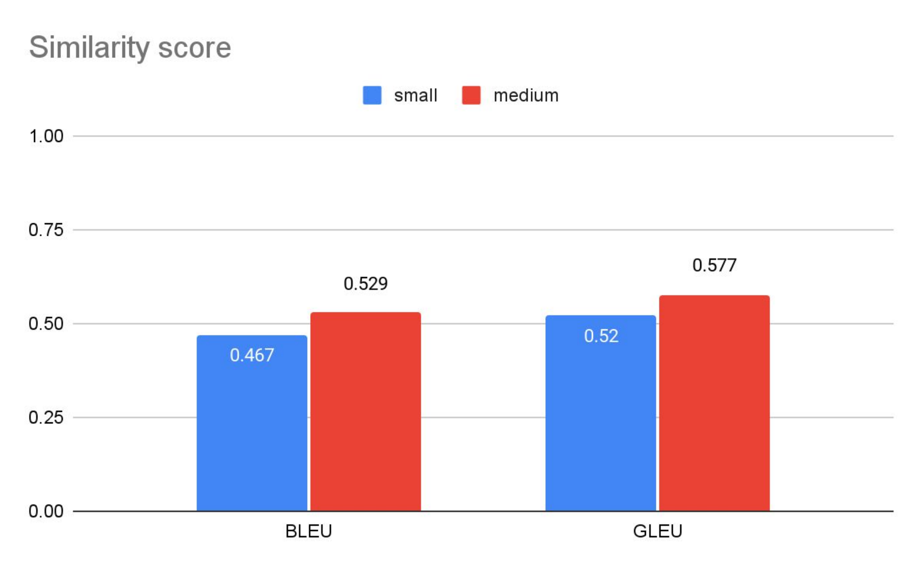
Evaluation method:

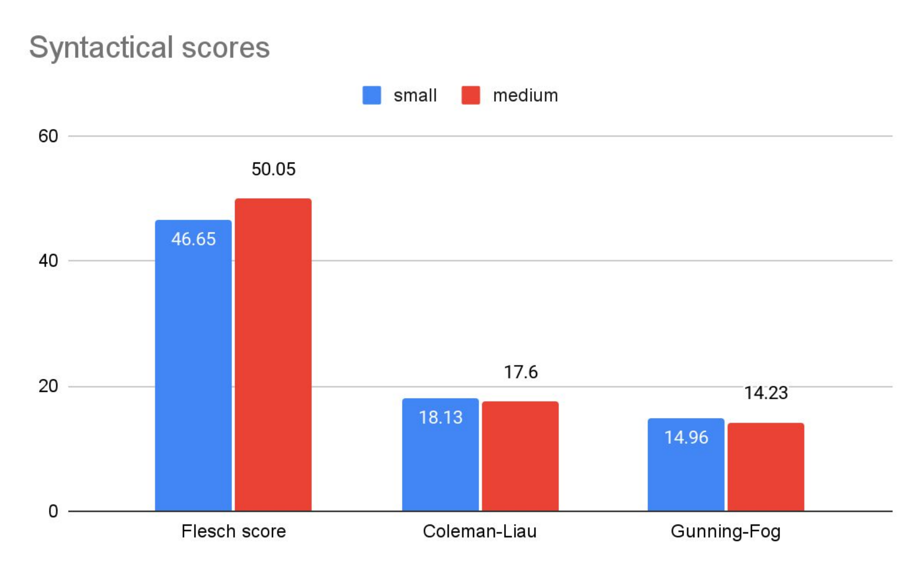
- test the trained model on test-dataset

Metrics for evaluation :

- Similarity scores:
  - BLEU
  - GLEU (Google BLEU)
- Syntactical scores:
  - Flesch reading ease score
  - Coleman-Liau index
  - Gunning-Fog index

Text Generation: baseline vs final model





Main focus:

- Improvement in both BLEU and GLEU performance
- Medium model presents more readable text compare to small model

GPT-2

Temperature parameter

Temperature :

In sequence generating models, one predicts the next token from distribution of the form:

$$softmax\left(\frac{x_i}{T}\right) \quad i = 1, 2 \ldots N$$

where *T* is the temperature

If the temperature is low, the model will probably output the most correct text, but rather boring, with small variation.

If the temperature is high, the generated text will be more diverse, but there is a higher possibility of grammar mistakes and generation of nonsense.

FLOWYGO

GPT-2

Temperature comparison

Lower temperature: around 0.5

The most advanced TV with advanced features. The all new and powerful HISENSE H55B7500 TV with advanced features makes it the perfect companion for your home. It comes with a powerful processor that provides the best 4K UHD experience. It comes with a powerful storage capacity and a powerful storage capacity. It is also equipped with Usb type-a Usb Connector Type. It is equipped with Audio Return Channel Arc. It comes with Parental Controls. It has Game Mode. It has Subtitles. It has High Dynamic Range Hdr 1000. It has Extended Pvr.

GPT-2

Temperature comparison
(HISENSE H55B7500 TV)

Higher temperature: around 0.8

Display your Android and iOS TV and its content conveniently using the H5500's remote, easily controlling media on it. The Remote is also compatible with smartphones and tablets and works well with most of them. The remote allows you to easy use your devices without any hassles. Besides, it is easy to pack and operate as it comes with USB type-a Usb Connector Type, Parental Controls and Usb type-a Usb Connector Type Cable.

Conclusions

Conclusions

summary

Named-entity recognition (NER):

- Increased dataset manually annotated
- Better and more stable model performance

Text-generation-model (GPT-2):

- The better NER model, The better the training data.
- Larger text generation model: GPT-2 medium
- New metric to evaluate the quality of generated text

Text-to-speech and text-to-video:

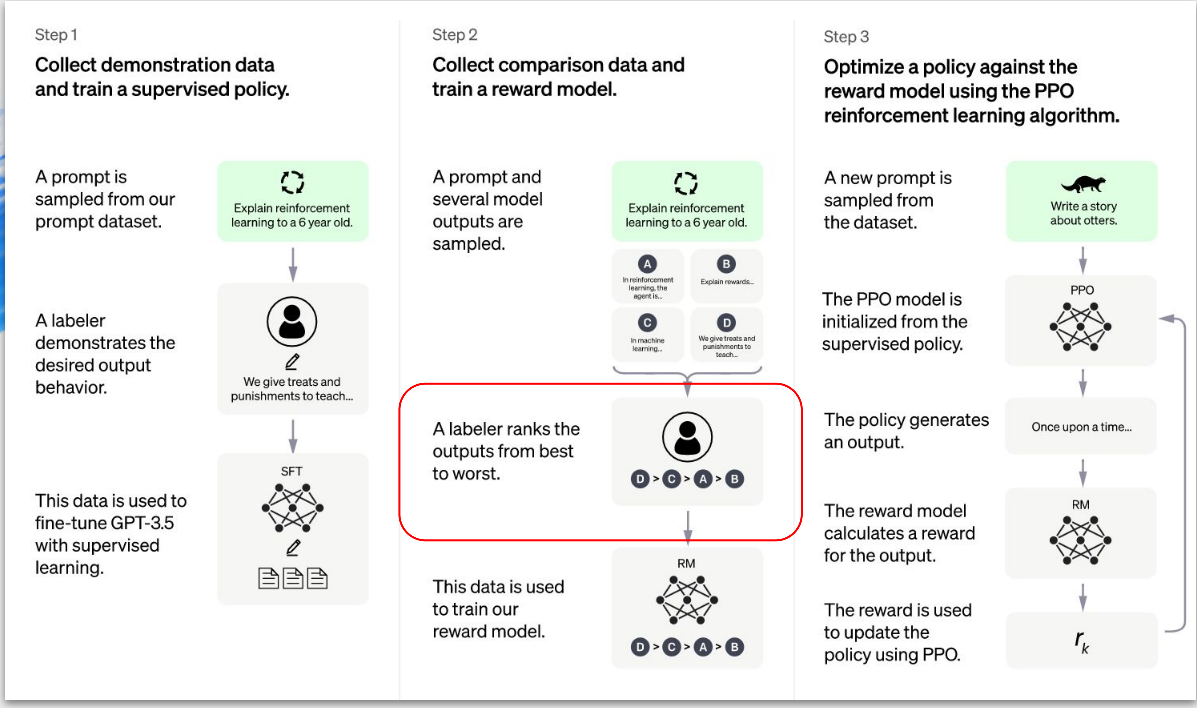
- Text-to-speech: gtts.
- Slides generation: PIL.
- Video generation: moviepy.
- The distance model.

- This system overall gives satisfactory results.

Conclusions

Evaluation always becomes a pain point:

- How good is the training dataset prepared by ner?
- How many people like the generated overview?
- How good is our video?



Step 1

Collect demonstration data and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3.5 with supervised learning.

Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

A new prompt is sampled from the dataset.

The PPO model is initialized from the supervised policy.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

Explain reinforcement learning to a 6 year old.
We give treats and punishments to teach...
SFT
Explain reinforcement learning to a 6 year old.
A
In reinforcement learning, the agent is...
B
Explain rewards...
C
In machine learning...
D
We give treats and punishments to teach...
D > C > A > B
RM
D > C > A > B
Write a story about otters.
PPO
Once upon a time...
RM
r_k

Hisense h55b7500

Hisense

Thanks you for your attention.
Questions?

Contacts:
Luca Agnese: lucaagnese@hotmail.it:
Matteo Donadio: matteodonadio00@gmail.com
Tianming Qu: qutianming0930@gmail.com
Flavio Spuri: spuri.flavio@gmail.com
Jiahao Zhang: giacomino99217@gmail.com