INF554

Data Challenge 2023
REPORT

# Extractive Summarization with Discourse Graphs

**By :**
Abasse DABERE
abasse.dabere@polytechnique.edu

Ivan TCHOMBA
ivan.tchomba-ngueko@polytechnique.edu

Mello DON
mello.don@polytechnique.edu

**Teacher :**
Vazirgiannis MICHALIS

# Contents

# 1 Introduction

This comprehensive report encapsulates the journey and achievements of our team in the 2023 INF554 Data Challenge. Set against the backdrop of this innovative competition, we embarked on a challenging yet enlightening endeavor to build an extractive summarization system for dialogue transcriptions. Our task involved the intricate analysis of 137 dialogues from a fictional electronics company, requiring us to meticulously identify and extract key utterances to form succinct, yet information-rich summaries. This challenge not only tested our technical prowess in natural language processing and graph machine learning but also offered a unique platform to explore the nuances of human dialogue and information extraction. Through this report, we aim to provide a detailed account of our methodologies, the technological innovations we employed, the obstacles we overcame, and the valuable insights we gained in the realm of dialogue summarization. This endeavor has been a blend of rigorous data analysis, creative problem-solving, and collaborative teamwork.

# 2 Metodology

## 2.1 Data Analysis

Our dataset is composed of dialogues in JSON format coupled with corresponding discourse graphs. This dual structure facilitates a thorough analysis of both the content and context of the dialogues.

• **Dialogue Analysis**: We began our analysis by examining word frequencies within the dialogues. This step helped us understand common linguistic patterns, including the prevalence of vocal sounds and idiomatic expressions. Here, the inclusion of the image would visually represent the most common words, providing a clear picture of the dialogue's linguistic landscape.

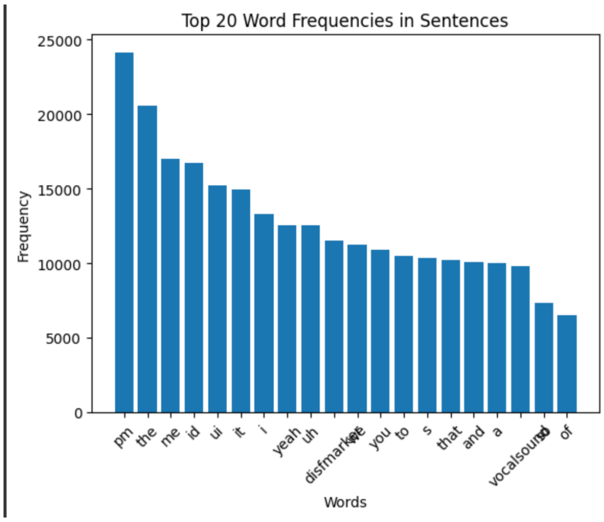
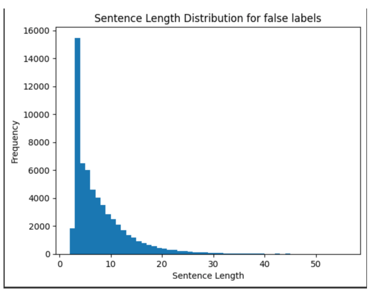
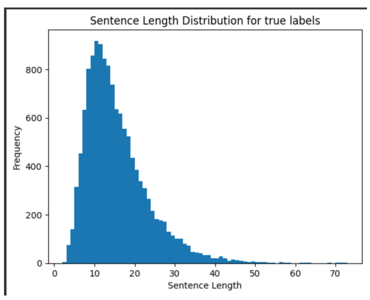



Figure 1: most frequent words

• **Sentence Length and Label Influence**: Next, we explored the length of sentences and its potential impact on the labeling process. Our findings, best illustrated by the following images for sentences with a 'False' label and for 'True' labeled sentences, showed notable variations in sentence length across different labels. These images would effectively convey how sentence length distribution varies with the labels.




(a) length distribution for False label sentences




(b) length distribution for True label sentences

• **Class Distribution**: The class distribution was then visualized to identify any imbalance in the dataset. The following image represent this distribution, showing the predominance of one class over the other. This imbalance informed our decision to use the F1 score as a crucial metric.

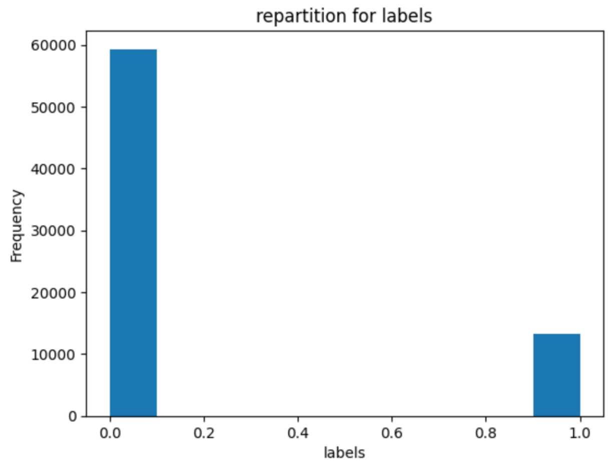



Figure 3: distribution across classes

• **Discourse Graph Analysis**: Lastly, we analyzed the most frequent attributes in our discourse graphs to understand their influence on the labeling process. The first image illustrate the common attributes in the discourse graphs. Furthermore, the two others provide a comparative view of attribute distribution across different labels.

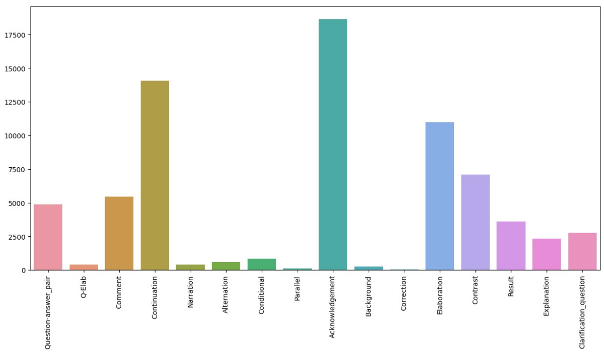
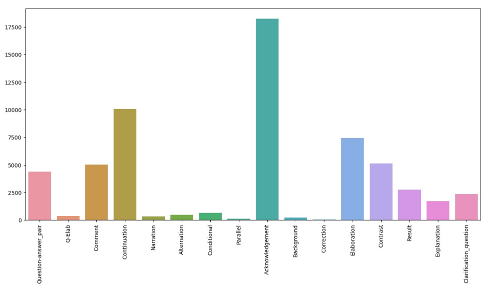
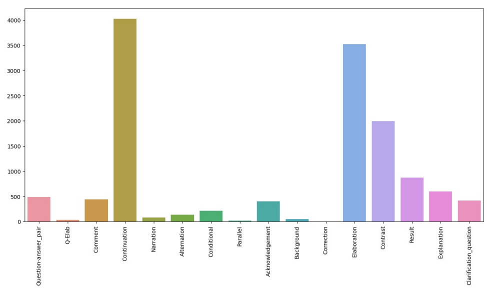



Figure 4: most frequent attributes



(a) most frequent attributes for True label sentences



(b) length distribution for True label sentences

## 2.2 Data preprocessing and features analysis

First, We then segmented the dialogues into individual utterances, each serving as a potential candidate for summary inclusion.

Following this, we start with cleaning the data, where we removed irrelevant elements such as non-verbal cues and background noise transcriptions, ensuring only meaningful text was retained.


Figure 6: Word cloud from the dataset

In the data cleaning phase, we utilized a word cloud to visualize the most frequent terms in our dataset. This analysis guided our cleaning strategy, leading us to eliminate vocal noises and expressions enclosed in "<" and ">". Subsequently, we removed colloquial fillers like "uh", "hum", and "actually", aiming to help our model concentrate on the most informative parts of the sentences. This process was pivotal in refining the dataset for more effective summarization by our model.

Once all of that done, we then encode our sentence with sentence transformer BERT to convert our text dataset in numerical one.

BERT (Bidirectional Encoder Representations from Transformers) is a groundbreaking model in natural language processing (NLP). Developed by Google, it revolutionized how machines understand human language. BERT's key innovation is its ability to analyze the context of a word in a sentence bidirectionally, rather than one-direction at a time. This allows for a more nuanced understanding of language, significantly improving performance in tasks like sentiment analysis, question answering, and language inference. BERT is pre-trained on a large corpus of text and can be fine-tuned for specific NLP tasks, making it highly versatile and effective.

## 2.3 Machine learning models

### 2.3.1 Best model

Our best model, a Deep Neural Network (DNN), stands out for its superior ability in predicting dialogue labels, as determined from the analysis of the provided notebook and context. Here's an in-depth look at how this model was developed and optimized:

- **Data Preprocessing**: The process began with comprehensive data preprocessing. We combined all dialogues into a substantial list of utterances, denoted as X. This text underwent a series of preprocessing steps, including lowercasing, tokenization, stop words removal, and stemming, which was critical in converting the dialogue text into a more analyzable form.

In addition to the dialogue text, we integrated discourse graph information into our model. We used graph attributes into prefixes and suffixes for the utterances, storing them in variables A (prefixes) and B (suffixes). This blend of dialogue texts and discourse graph attributes was vital in capturing the full context of the conversations.

The dataset Z, consisting of X, A, and B, was split into training and validation sets. We employed hot-one encoding on A_train and B_train to transform these into tabular data, which was then used to encode A_valid and B_valid.

• **DNN Architecture and Iterations**: Our DNN model, renowned for its effectiveness in handling complex data patterns, was subjected to numerous architectural refinements. Throughout these iterations, we focused on optimizing the model's structure to best interpret the intricate relationships in our data.

After testing multiple architectures, we settled on the one that yielded the highest F1-score. This approach was not only methodical but also highly effective, as the best architecture achieved an impressive F1-score of 59%. This score was a key indicator of the model's ability to balance precision and recall, especially in the context of our unevenly distributed class data.

• **Model Training**: During the training phase, we placed significant emphasis on addressing the class imbalance within the dataset. This was crucial for ensuring that the model did not develop a bias toward the more prevalent class and could generalize well across all classes. The training process was closely monitored, with performance metrics guiding iterative improvements.

• **Performance Evaluation**: The model's evaluation on the validation set employed key metrics like accuracy, precision, recall, and the pivotal F1 score. Our DNN demonstrated exceptional performance, notably outperforming other models in accurately predicting the labels of dialogues.

The selected DNN model, with its meticulously optimized architecture and thorough training that accounted for class imbalance, emerged as the top performer in our analysis. It not only delivered the highest F1-score but also demonstrated a profound ability to navigate the complexities of dialogue data. Its robustness and adaptability mark it as the ideal choice for our project, capable of providing insightful and reliable predictions in conversational dynamics.

### 2.3.2 Other tested models: Random Forest

In our experimental approach, following data preprocessing, we experimented with various models for the desired classification. We started with simpler models, gradually increasing complexity. Since decision trees were used in the baseline, we began with techniques like Random Forest, particularly using the AdaBoost model. As Random Forest tools do not utilize GPUs, the computation was slow, limiting our ability to test numerous architectures. We varied the number of weak learners from 50 to 400 and observed the outcomes. Our best score on the development set was 49%, achieved with a dataset encoded using BERT, augmented with sentence relationships in One Hot encoding form. Following that, we experimented with other Random Forest algorithms like Bagging and XGBoost. However, at best, we achieved only marginally better performance. This led us to pivot our approach and explore different avenues, particularly focusing on neural network-based methods.

#### 2.3.3 Graph Convolutional networks

article graphicx

# 3 Graph-based Approach

The objective was to leverage the capabilities of graph neural networks to capture intricate relationships within dialogue data, presenting additional contextual information. This data augmentation aimed to enrich our comprehension of the associations between utterances and their contextual significance.

## 1. Graph Construction

1. Utterances were encoded to form nodes, and relationships between phrases constituted the edges.

2. Encoding of relationships utilized the ‘OneHotEncoding‘ class from the ‘scikit-learn‘ library.

The resulting graph encapsulated both utterance nodes and their interconnectedness based on encoded relationships.

## 2. Architecture Design

The subsequent critical phase revolved around devising an optimal architecture, focusing on Graph Convolutional Networks (GCN) implemented through PyTorch Geometric.

#### 3.0.1 Choice of Graph Convolutional Layer

The selection of an appropriate GCN layer was pivotal. It involved considering various options, such as traditional GCNs, GraphSAGE, or Gated Graph Neural Networks (GGNN), based on the problem complexity and specific relationship nuances among utterances.

#### 3.0.2 Layer Stacking Strategies

Layer stacking played a vital role in capturing and representing complex relationships. Experimentation with multiple layers was conducted to gauge their effectiveness in capturing nuanced relationships while avoiding information overload or loss.

#### 3.0.3 Incorporation of Additional Features

Beyond the core relationships among utterances, the integration of supplementary features or information was explored to enrich the graph representation. This involved considering additional node attributes, contextual data, or syntactic information.

#### 3.0.4 Activation Functions and Regularization

The choice of activation functions and the application of regularization techniques were pivotal to prevent overfitting and improve model generalization. Mechanisms such as dropout or specific activation functions were employed to enhance model performance.

#### 3.0.5 Training and Evaluation

With the defined architecture, model training utilized annotated data, while evaluation involved employing appropriate metrics like accuracy, recall, or F-score, tailored to the nuances of the utterance relationship problem.

The implementation of this architecture in PyTorch Geometric involved defining layers, managing graph-structured data, and implementing training and evaluation phases using functionalities provided by this library.

This approach effectively harnessed the power of Graph Convolutional Networks to capture and comprehend intricate relationships between utterances, significantly enhancing our understanding of dialogue context and the nuanced significance of interactions within the project's scope.

However, despite exploring various models and investing efforts into optimizing the architecture, we encountered challenges in improving upon existing models. The difficulty in effectively modeling the exchange of information between nodes and edges within our graph limited our ability to achieve superior results compared to other models. This underscores the complexity inherent in capturing the nuanced relationships in dialogue data and suggests avenues for future research and refinement of graph-based approaches.

# 4 Conclusion

In conclusion, our exploration into the realm of machine learning for dialogue analysis has been both challenging and enlightening. The 'inf554-extractive-summarization-2023' project presented a unique opportunity to delve deep into the nuances of natural language processing and understand the complexities involved in extractive summarization.

Through rigorous data preprocessing and exploration of various machine learning models, from simpler ones like DecisionTrees to more sophisticated Deep Neural Networks (DNN), we have gained invaluable insights into the nature of dialogue data and the intricacies of predictive modeling. Each model tested provided us with crucial learning points, helping us to understand the limitations and strengths of different approaches in handling imbalanced datasets and complex relational structures.

The standout performer, our DNN model, exemplified the power of advanced machine learning techniques in tackling complex tasks. Its superior F1-score not only demonstrated its effectiveness in making accurate predictions but also underscored the importance of choosing the right model architecture and training approach for specific types of data.

This project also highlighted the significance of thorough data preparation. The combination of textual data with discourse graph attributes played a pivotal role in enhancing the models' understanding of the dialogues, showcasing the importance of feature engineering in machine learning.

As we conclude this report, it's clear that the journey through this project has been as important as the destination. The lessons learned extend beyond the confines of this specific task, providing broader insights into the field of machine learning and its application in natural language processing. The experience and knowledge gained lay a strong foundation for future endeavors in this vibrant and ever-evolving domain.