deti universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

Voice-Controlled Excel Management System

Multimodal Interaction - 1ºAssigment

Trabalho realizado por:
Carolina Silva - 113475
Tomás Brás - 112665

2025/2026

Índice

1. Introduction

This document presents the architecture and implemented functionalities of a voice-controlled system designed to manipulate and analyse data within Microsoft Excel. The system enables natural speech interaction, allowing users to perform structured operations such as calculating averages, applying conditional formatting, analysing improvement metrics, inserting columns and questions, generating charts, updating grades, creating pivot tables, removing visual elements, executing statistical queries, and saving files through voice commands.

The Natural Language Understanding (NLU) component is built using Rasa 3.1. It combines lookup tables, synonym dictionaries, regular expressions, and a refined set of entities to identify student names, student numbers, test identifiers, question ranges, column labels, regimes, and numerical values. The DIETClassifier, enhanced with regex and lookup-table support, provides robust entity extraction even in the presence of variable or unstructured academic terminology.

System behaviour is governed by a deterministic rule-based action framework. Each intent is mapped to a corresponding action through RulePolicy configurations and story definitions, ensuring predictable and reliable execution for tasks such as generating charts, inserting questions, updating grades, or constructing pivot tables. A fallback mechanism is included so that when the NLU confidence score falls below the configured threshold of 0.4, the assistant triggers the default fallback action and provides corrective verbal feedback.

Every interaction produces spoken output through predefined utterance templates, ensuring clarity and traceability of all operations performed on the Excel file. Overall, the system demonstrates how natural language interfaces can enhance accessibility, reduce the cognitive load associated with complex spreadsheet manipulation, and support users with different levels of digital literacy.

2. Functionalities and Implemented Intents

Below is a detailed description of each intent's purpose, examples, and implementation approach, fully updated according to the latest NLU structure.

2.1. Greetings & Conversational Intents

2.1.1 Greet

Function: The *greet* intent allows the assistant to recognize user greetings and initiate a conversational interaction.

Example Commands:

- "Olá Assistente"
- "Bom dia"
- "Olá, tudo bem?"

System Feedback:

- "Olá! Como posso ajudar?"

2.1.2 Ask how are you

Function: The *ask how are you* intent lets the user ask about the assistant's state.

Example Commands:

- "Como estás?"
- "Tudo bem contigo?"

System Feedback:

- "Estou aqui e pronto para ajudar!"

2.1.3 Respond how am I

Function: The *respond how am I* intent captures the user's answer to "How are you?".

Example Commands:

- "Estou bem"
- "Estou cansado"

System Feedback:

- "Percebi. Vamos continuar!"

2.1.4 Helper

Function: The *helper* intent provides a summary of available features.

Example Commands:

- "Que comandos posso usar?"

- "Mostra a lista de funcionalidades"

System Feedback:

- "Posso calcular médias, criar gráficos, gerar pivots e muito mais."

2.1.5 Fallback

Function: Triggered when the system cannot classify the user command.

Examples: “Não percebi.”

Implementation: Executes utter_default.

2.2. Excel Manipulation

2.2.1 Calculate Average

Function: Enables the assistant to compute individual or class-wide averages.

Entities used: aluno_nome, aluno_numero

Example Commands:

- "Calcula a média da turma"
- "Qual é a média da aluna Maria?"

System Feedback:

- "Média calculada."
- "Aluno não encontrado."

Implementation:

action_calcular_media applies or updates Excel formulas, ensuring correct column detection.

2.2.2 Highlight Approved/Reproved

Function: Marks students visually according to performance.

Example Commands:

- "Destaca os alunos aprovados e reprovados"
- "Pinta a verde quem passou"

System Feedback:

- "Situação atualizada."

Implementation: Formats the Média column using thresholds.

2.2.3 Insert Columns

Function: Creates new columns such as Situação, Comentários, or Teste-related columns.

Entities: coluna_nome, teste_numero, coluna_teste

Example Commands:

- "Insere a coluna Situação"
- "Cria a coluna Comentários"

System Feedback:

- "Coluna criada com sucesso."

Implementation: Dynamically adds columns in Excel at the correct index.

2.2.4 Insert Questions

Function: Adds question columns (e.g., T2_P1–P5) dynamically.

Entities: teste_numero, pergunta (supports ranges: P1–P5)

Example Commands:

- "Adiciona perguntas ao teste 2"
- "Cria as perguntas P1 a P5 no teste 3"

System Feedback:

- "Perguntas adicionadas."

Implementation: Creates columns for questions and registers them.

2.2.5 Update Grades

Function: Updates values for individual questions, full tests, zeroing, randomization, or entire class operations.

Entities: pergunta, teste_numero, aluno_numero, aluno_nome, valores

Example Commands:

- "Atualiza a pergunta 2 do teste 1 para 4"
- "Define valores aleatórios para o teste 3"

System Feedback:

- "Notas atualizadas."

Implementation:
action_atualizar_notas handles all modes, adjusting formulas when necessary.

2.2.6 Real Improvement

Function: Calculates each student's improvement between tests, adding color-coded results.

Example Commands:

- "Quem melhorou?"
- "Mostra evolução entre testes"

Implementation: Computes progression and updates “Melhoria Real”.

System Feedback:

- "Melhoria Real calculada."

2.2.7 Possible Improvement

Function: Identifies which students can still reach a positive mark and what grade is needed.

Example Commands:

- "Quem tem MP?"
- "Quem ainda pode recuperar?"

System Feedback:

- "Melhoria possível calculada."

Implementation: Uses “Nota Necessária” and “Situação MP” logic.

2.2.8 Mathematical Operations

Function: Provides statistics: number of passes, averages, medians, high performers, etc.

Example Commands:

- "Quantos aprovados existem?"
- "Mostra estatísticas gerais da turma"

System Feedback:

- "Tabela de estatísticas criada."

Implementation: Computes metrics and can generate summary tables.

2.2.9 Save File

Function: Stores the updated Excel file with a chosen filename.

Entity: nome_ficheiro

Example Commands:

- "Guarda o ficheiro como Resultados.xlsx"
- "Salva o documento"

System Feedback:

- "Ficheiro guardado."

Implementation: Performs file-safe save operation.

2.3. Charts

2.3.1 Chart Class Performance

Function: Generates a bar chart comparing Test 1, Test 2, and Average.

Entities: grafico

Example Commands:

- "Gera gráfico da turma"
- "Mostra a evolução da turma"

System Feedback:

- "Gráfico criado.

Implementation: Uses line, bar, or evolution charts.

2.3.2 Student Bar Graph

Function: Creates a bar chart showing a single student's performance.

Entities used: aluno_nome, aluno_numero

Example Commands:

- "Gera gráfico de barras do aluno 123456"
- "Mostra gráfico da aluna Maria"

System Feedback:

- "Gráfico criado."

2.3.3 Test Question Chart

Function: Builds a graph of averages for T2_P1 to T2_P5. **(The columns T2_P1–P5 are detected automatically.)**

Example Commands:

- "Gera gráfico das perguntas do teste 2"
- "Mostra o gráfico das T2_P"

System Feedback:

- "Gráfico criado."

Implementation: Fixed-column average comparisons.

2.3.4 Delete All Chart

Function: Deletes every chart from the sheet.

Example Commands:

- "Apaga todos os gráficos"
- "Remove os gráficos"

System Feedback:

- "Gráficos apagados."

Implementation: Deletes every chart object.

2.4. Pivot Tables

2.4.1 Create Pivot Table

Function: Creates dynamic pivot tables with optional rows, values, and filters.

Entities used: coluna_excel_row, coluna_excel_value, regime

Example Commands:

- "Cria uma tabela dinâmica"
- "Quero Nome nas linhas e REGIME nos valores"
- "Pivot filtrada pelo regime TP"

System Feedback:

- "Tabela dinâmica criada."

3. Entities, Lookup Tables and Regex Patterns

To support high-precision extraction, the system employs multiple entity types.

3.1. Entities

- **aluno_nome**: full Portuguese names (regex + lookup)
- **aluno_numero**: mechanographic numbers (4–12 digits)
- **valores**: numeric values, including multiple values
- **pergunta**: P1–Pn, with support for ranges
- **teste_numero**: identification of test numbers
- **coluna_nome**: dynamic column creation
- **coluna_teste**: column representing a whole test
- **coluna_excel**, **coluna_excel_row**, **coluna_excel_value**: fields used for PivotTables
- **regime**: filters for pivot (O, TE - Trabalhador Estudante, TP- Tempo Parcial)
- **grafico**: keywords for chart types
- **nome_ficheiro**

3.2. Lookup Tables

Lookup tables significantly enhance the system's ability to recognise domain-specific vocabulary with high accuracy. They provide predefined lists of terms that frequently appear in user commands, reducing ambiguity and supporting consistent entity extraction.

In this project, lookup tables were essential for:

- **Student names**
  Allowing the recognition of common Portuguese names and full name structures, ensuring reliable identification even when multiple students share similar first names.

- **Test and column names**
  Including fixed spreadsheet labels such as *"Teste 1"*, *"Média"*, *"T2_P1"*, *"Situação MP"*, and others, enabling the system to correctly map natural-language references to actual Excel columns.

- **Regimes**
  Supporting multiple linguistic variations (e.g., *"O"*, *"ordinário"*, *"TP"*, *"tempo parcial"*, *"trabalhador"*), ensuring robust classification regardless of user phrasing.

- **Row and value fields for pivot tables**
  Providing controlled vocabularies that distinguish between:

  - fields used as **rows** (e.g., *Nome*, *REGIME*, *Situação MP*)

  - fields used as **values** (e.g., *Média*, *Nota Nec*, *Teste 2*)
    This was crucial for avoiding errors in pivot table construction.

3.3. Synonyms

This was particularly important for **pivot table generation**, where column references must match the exact Excel header names.

Synonyms were used to normalise:

- **Column names**
 (e.g., *“Nome”*, *“o nome”*, *“nome do aluno”*, *“REGIME”*, *“tipo de regime”*, *“Média”*, *“a média”*)
 → Ensuring that all variations map to the exact spreadsheet column label required for building a pivot table.

- **Test names**
 (e.g., *“teste 1”*, *“t1”*, *“teste um”*, *“teste nº1”*)
 → Avoiding mismatches between spoken expressions and the actual column names like *“Teste 1”* in Excel.

- **Regimes**
 (e.g., *“O”*, *“ordinário”*, *“Ordinário”*, *“tp”*, *“tempo parcial”*)
 → Allowing pivot table filters to apply correctly, even when users use different linguistic forms.

Because pivot tables are extremely sensitive to exact field names, any small mismatch (uppercase/lowercase, accents, additional words) would result in errors or invalid table construction. The synonym dictionaries ensured that the NLU always produced **clean, normalised entity values** compatible with the spreadsheet structure.

3.4. Regex

They enabled precise extraction of entities even when the user spoke in a natural, free-form manner.

These patterns were crucial for operations such as updating grades, generating graphs, and inserting questions.

The regexes covered:

- **Complex full names**
 Recognising multi-part Portuguese names with accents, hyphenation, or multiple surnames (e.g., *“Maria Inês Oliveira Silva”*), ensuring the correct student is identified.

- **Number sequences for grades**
 Extracting lists like *“2 4.5 5 3 1”*, *“1, 2, 3, 4, 5”*, or mixed formats, which are required when updating multiple questions at once.

- **Question ranges**
 Understanding expressions such as *“P1 a P5”*, *“pergunta 2 até 6”*, or *“questão 3–8”*, enabling automatic generation or updating of whole blocks of column labels.

- **Test numbers**
 Interpreting phrases like *“teste 1”*, *“t 2”*, or *“teste número 3”*, which are needed for calculating progress, inserting questions, and performing grade operations.

- **Chart keywords**
  Detecting terms such as *“barras”*, *“evolução”*, *“desempenho”*, etc., to classify the requested chart type correctly.

These ensure extraction works even in natural, unstructured speech.

4. Rules and Stories

Rules define deterministic behaviours such as:

- generating charts
- saving files
- computing averages
- inserting columns

Stories support multi-turn conversational flows when the system requires additional context or confirmation.

5. System Usage Instructions

1. Train Rasa: $ rasa train
2. rasa run --enable-api -m ./models --cors "*"
3. To test the model **rasa shell nlu**
4. Start the multimodal stack:
   - Rasa Server
   - Fusion Engine
   - MMI Framework
   - Voice detection webpage
5. Run the Excel-backend application.
6. Speak a command using any of the supported intents.
7. Observe updates on the Excel sheet and feedback from the assistant.

6. *Representative example command for each intent*

- **helper** — “Mostra a lista de comandos.”
- **calcular_media** — “Calcula a média da turma.”

  — Calcula a média da aluna Maria Silva”
- **inserir_colunas** — “Insere a coluna Situação.”
- **destacar_aprovados_reprovados** — “Destaca os alunos aprovados e reprovados.”
- **melhoria_real** — “Mostra quem melhorou.”
- **melhoria_possivel** — “Mostrar MP.”
- **gerar_grafico_turma** — “Gera gráfico de evolução da turma.”
- **gerar_grafico_barras_aluno** — “Faz gráfico de barras para o aluno número 123456.”
- **gerar_grafico_perguntas_t2** — “Gera gráfico das perguntas do teste 2.”
- **apagar_todos_graficos** — “Apaga todos os gráficos.”
- **guardar_ficheiro** — “Guarda o ficheiro como Relatorio_Final.xlsx.”
- **operacoes_matematicas** — “Gera estatísticas gerais da turma”
- **criar_pivot_table** — “Cria uma tabela dinâmica.”

7. Conclusion

The Voice-Controlled Excel Management System shows how natural language can make Excel more dynamic, accessible, and intuitive, especially for users with limited digital literacy. The project expanded Excel's analytical capabilities through voice-driven operations such as pivot table generation, column and question insertion, grade manipulation, and improvement analysis, while also improving understanding of Excel automation, regex extraction, entity modelling, and Rasa's NLU workflow.

Through the combination of lookup dictionaries, refined entities, deterministic rules, and confidence-based behaviour, the assistant achieves accurate and reliable interpretation of spoken commands. The system proves that voice interaction can simplify complex spreadsheet tasks and reduce errors, making Excel more inclusive and easier to use. In the future, additional functionalities and improved conversational flows can further enhance accessibility and usability.

8. References

- https://learn.microsoft.com/en-us/dotnet/csharp/advanced-topics/interop/how-to-access-office-interop-objects
- Lecture demonstration videos
- Rasa documentation: https://rasa.com/docs/rasa/nlu-training-data