Universidade de Aveiro Departamento de Comunicação e Arte
2018

Diogo Filipe Guerreiro Oliveira

Interação por Linguagem Natural com o MEO: a abordagem ALICE

Universidade de Aveiro – 2018
Departamento de comunicação e arte (DeCA)

Diogo Filipe Guerreiro Oliveira

Interação por Linguagem Natural com o MEO: a abordagem ALICE

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Comunicação Multimédia, realizada sob a orientação científica do Dr. Jorge Ferraz de Abreu, Professor Auxiliar c/ Agregação do Departamento de Comunicação e arte da Universidade de Aveiro, e do Engenheiro Herlander Santos da Altice Labs.

Projeto de dissertação para obtenção do grau académico no âmbito do Programa de Bolsas de Investigação da Inova-Ria – Programa GENIUS, autorizado pela FCT – Bolsa de Iniciação Científica (BIC) financiada pela Inova-Ria e desenvolvido na Empresa Altice Labs

Dedico este trabalho todos os que me acompanharam nesta etapa, em especial à minha família que me permitiu alcançar esta meta e que sempre me ajudou a orientar os meus objetivos, aos quais dedico também o meu mérito, bem como aos docentes que me guiaram ao longo deste processo e a todos os colaboradores da Altice Labs que me apoiaram durante o estágio.

o júri

presidente

Professor Doutor Nelson Zagalo
Professor Associado, Universidade de Aveiro

Professora Doutora Alcina Maria Narciso Prata
Professora Adjunta, Instituto Politécnico de Setúbal

Professor Doutor Jorge Trinidad Ferraz de Abreu
Professor Associado C/ Agregação, Universidade de Aveiro

agradecimentos

Sinto-me grato por todo o apoio que me foi dado pela minha família, amigos e docentes. Em especial, quero agradecer ao meu avô Armando pelo seu apoio incansável, que nunca me faltou com nada para que eu pudesse obter o grau de Mestre, à minha namorada, aos meus pais e ao meu irmão por me apoiarem nos momentos em que me senti mais frágil. Obrigado por terem feito parte do meu desenvolvimento pessoal, ajudando-me a manter o foco necessário para o desenvolvimento desta dissertação bem como do estágio profissional na Altice Labs.

palavras-chave

Linguagem Natural, televisão interativa, chat bot, natural language understanding, personalização de conteúdos

resumo

Por longos anos a televisão tem sido uma fonte de entretenimento relaxante para toda a família. Com a evolução da tecnologia, a televisão passou a ser transmitida pela internet (IPTV), o que permitiu o acesso a conteúdos através de uma vertente mais interativa, bem como um aumento significativo de funcionalidades que proporcionam uma oferta de conteúdos *on-demand*. Através de novas funcionalidades como o EPG, as gravações automáticas e o videoclube, são proporcionados ao utilizador um conjunto de informação sobre os conteúdos disponíveis de forma linear (EPG), mas também o acesso a conteúdos a pedido (como o videoclube e as gravações automáticas). No entanto, interagir com estas funcionalidades via controlo remoto requer um número significativo de cliques até se encontrar o que se pretende.

A investigação aqui descrita enquadra-se no desenvolvimento de uma aplicação de televisão interativa, para aceder a conteúdos personalizados, através de linguagem natural. Entende-se por linguagem natural aquela que os humanos utilizam para comunicar. Pretende-se desenvolver uma solução tecnológica que, recorrendo à interação por linguagem natural, seja possível marcar conteúdos para ver mais tarde, seguir séries e agendar conteúdos para que possa ser notificado mais tarde. Para além disso, pretende-se que sempre que o utilizador ligue a televisão seja notificado de conteúdos que deixou a meio, na última vez que esteve a ver televisão, quando alugou um filme no videoclube ou que marcou para ver mais tarde e vão expirar.

O protótipo desenvolvido foi feito em contexto empresarial na Altice Labs e é constituído por um conjunto de aplicações que gerem o estado da interação por linguagem natural e comunicam com o sistema televisivo MEO, enriquecendo a experiência televisiva através uma interação mais natural para o ser humano.

keywords

Natural language, interative television, chat bot, natural language understanding, content personalization

Abstract

For many years the television has been a source of relaxing entertainment for the whole family. With the evolution of technology, television started to be transmitted over the Internet (IPTV), which allowed access to content through a more interactive side, as well as a significant increase of features that provide an offer of on-demand content. Through new features such as EPG, automatic recordings and videoclub, the user is provided with a set of information on the available content in a linear way (EPG), but also access to content on demand (such as the video store and automatic recordings). However, interacting with these features via remote control requires a significant number of clicks until you find what you want.

The research described here is part of the development of an interactive television application to access personalized content through natural language. It is understood by natural language that which humans use to communicate. It is intended to develop a technological solution that, using natural language interaction, can mark content for later viewing, follow series and schedule content so that it can be notified later. In addition, it is intended that whenever the user turns on the television to be notified of content that he left in the middle, the last time he was watching television, when he rented a movie in the video store or that he checked to watch later, and they will expire.

The prototype was developed in a business context in Altice Labs and consists of a set of applications that manage the interaction state through natural language and communicate with the MEO television system, enriching the television experience through a more natural interaction for the human being.

Índice

Índice de Figuras

Índice de tabelas

Índice de gráficos

Índice de anexos

Acrónimos

API – *Application Programming Interface*

ASR – *Automated Speech Recognition*

CHIC – *Cooperative Holistic View on Internet and Content*

DRM - *Digital Rights Management*

DVR – *Digital Video Recorder*

EPG – *Electronic Program Guide*

GA – Gravações automáticas

GUID – *Globally Unique Identifier*

ILN – Interação por Linguagem Natural

IP – *Internet Protocol*

IPTV – *Internet Protocol Television*

NLG – *Natural Language Understanding*

NLP – *Natural Language Processing*

NLU – *Natural Language Understanding*

SQL – *Structured Query Language*

SR – Sistemas de Reconhecimento

SRP – *Speech Recognition Program*

STB – *set-top-box*

TTS – *text-to-Speech*

TV – Televisão

UI – *User Interface*

VoD – *Video on Demand*

I.Introdução

Com a evolução da transmissão de sinal de televisão por IP, surgiram novas funcionalidades que mudaram o modo como os utilizadores interagem com a televisão, transpondo a mudança de canais com o telecomando para uma vertente mais interativa, onde os conteúdos são os elementos centrais. Desta forma, é necessário refletir sobre como é que a interação com a televisão (TV) pode evoluir para conseguir acompanhar os avanços tecnológicos, visto que as aplicações televisivas têm cada vez mais funcionalidades que precisam de muitos cliques de botões de um telecomando para serem utilizadas. Desta forma, surge uma necessidade de adaptar a interação com a televisão, para permitir a utilização de funcionalidades mais avançadas, ou que requerem muitos cliques, de uma forma que lhe seja mais natural, como o caso da voz. O tipo de interação por voz existente no mercado das telecomunicações, permite apenas o uso de comandos simples e curtos em formato robotizado, uma vez que só existe um pedido válido para a execução de cada tarefa, levando a um aumento de esforço cognitivo no utilizador para memorizar os comandos. No entanto, com a evolução da interação por linguagem natural (ILN), soluções de mercado como as da Amazon e da Google começam a ser estendidas para as *SmartTV's* e *Media Centers*, tornando possível a criação de sistemas conversacionais Humano-Computador, que sejam capazes de criar e manter um discurso contínuo, refinando o resultado da pesquisa, mediante sucessivas iterações do utilizador com um léxico de palavras mais abrangente. Desta forma, podemos executar comandos e pesquisas mais complexas, melhorando a qualidade da comunicação por voz e criando maior empatia com quem comunica com o sistema.

1.Objetivos e finalidades do trabalho

O principal objetivo deste projeto é a conceção, construção e implementação de um protótipo de um sistema conversacional através da interação por linguagem natural, permitindo ao utilizador interagir por voz a fim de solicitar a continuação do visionamento de conteúdos televisivos que, já tendo sido iniciados não tenham sido finalizados.

Para implementar este sistema, é necessário começar por analisar como os utilizadores perspetivam a interação com a televisão por linguagem natural e definir através da análise dos dados, quais as frases e palavras mais recorrentes para interagir com os conteúdos. Para além disso, é necessário fazer um mapeamento das tecnologias que compõem a interação por linguagem natural, de forma a definir quais são as mais adequadas para esta implementação.

A implementação do protótipo consistirá em duas partes, sendo a primeira fase o treinamento do sistema, mais concretamente a componente de *Natural Language Understanding permitindo* compreender os comandos neste contexto específico da televisão e a integração deste sistema conversacional com a aplicação MEO[1]. Depois da fase de implementação estar concluída, será necessário realizar uma avaliação do protótipo, para compreender o contributo da ILN no contexto do ecossistema televisivo.

[1] Aplicação televisiva do operador MEO, do grupo Altice.

2.Metodologia

O processo de investigação teve como questão inicial como é que os utilizadores gostariam de interagir com a televisão através de interação por linguagem natural. Fruto desta fase preliminar de reflexão, surgiu a questão de investigação: "Como recorrer à Interação por Linguagem Natural para guardar diferentes tipos de conteúdos e ativar um *reminder* televisivo?" Com esta questão, pretende-se solucionar um problema atual relativo à interação com a televisão. Como é referido por Coutinho (2011, pag. 6):

> *"Investigar é assim uma atividade que pressupõe algo que é investigado, uma intencionalidade de quem investiga e um conjunto de metodologias, métodos, e técnicas para que a investigação seja levada a cabo numa continuidade que se inicia com uma interrogação e termina com a apresentação pública dos resultados da investigação"*

Como é possível compreender pelo enquadramento que se encontra no capítulo, a interação através de simples comandos por voz não é a solução mais eficaz para satisfazer todos os pedidos que podem ser executados pelo tradicional telecomando, comprovando a necessidade de encontrar um mecanismo mais robusto, como a interação natural por voz.

A metodologia sobre a qual assenta o presente estudo é a investigação de desenvolvimento (Oliveira, 2006), pois pretende-se analisar como é que os utilizadores gostariam de interagir por voz, com a televisão através de linguagem natural, para implementar um protótipo que seja de fácil utilização, capaz de entender um pedido de continuar a ver um conteúdo, de diferentes formas. Esta metodologia subdivide-se em três tipos: desenvolvimento de conceito, desenvolvimento de objeto e aperfeiçoamento de habilidades pessoais enquanto utensílios profissionais (Van der Maren, 1996, p. 178). A investigação de desenvolvimento enquadra-se no segundo tipo, no desenvolvimento de objeto, uma vez que este pretende solucionar problemas a partir da prática quotidiana e também, apresentar propostas de princípios de design para futuras iterações, através da conceptualização, construção e implementação de um protótipo (Van der Maren, 1996, p. 179-180). Segundo Van Der Maren, tal como citado por Lia Oliveira (Oliveira, 2006, p.71), existem duas finalidades neste modelo, sendo a primeira a de criticar, contestar e colocar em questão a forma de pensamento geral, partindo do pressuposto de que a verdade absoluta não existe. A segunda finalidade remete para a procura de novo conhecimento, ideias e hipóteses, através da contravenção dos saberes admitidos, regendo-se pelas normas e pela ética da atividade científica. No entanto, segundo os autores Rita C. Richey e James D. Klein (Richey & Klein, 2005) existem apenas duas formas, definidas como tipo um e dois. O tipo um de metodologia de desenvolvimento diz respeito aos estudos de desenvolvimento que visam o interesse em recomendar ou identificar princípios gerais, sobre um determinado produto, aplicação ou ferramenta. Como tal, este é aplicado para o estudo de design de produto, desenvolvimento e avaliação. Já o tipo dois de metodologia de desenvolvimento foca-se em validar modelos de design e processos, identificando as condições de sucesso para a sua aplicação. Assim, este estudo enquadra-se no tipo um, visto que a conclusão que se vai obter pode depender do público específico que vamos endereçar para realizar a experiência e tem um foco mais na parte de desenvolvimento do protótipo, pelo que no

final se pretende validar a facilidade de acesso a conteúdos com suporte de um sistema conversacional com interação por voz, através de linguagem natural.

2.1. Participantes do estudo de caso

Pretendem-se analisar dois grupos de participantes, que vão ser angariados através de uma amostra por conveniência. Estes participantes não representam o universo em estudo, pelo que os resultados apenas correspondem a dois públicos específicos de utilizadores. Destes elementos, pretende-se que oito correspondam a pessoas que já interagiram por voz com uma aplicação e os restantes sete que não possuam experiência de utilização com este tipo de interação.

2.2. Técnicas e Instrumentos de recolha de dados

Para esta investigação, serão aplicados vários métodos qualitativos de recolha de informação, para avaliar o protótipo desenvolvido, aplicando uma estratégia de pesquisa que compreende um método que abrange a análise de dados, o estudo de caso (Yin, 2001). O estudo de caso do tipo exploratório aplica-se para esta investigação, uma vez que se pretende identificar questões de pesquisa e possíveis abordagens para projetos futuros. Para esta finalidade serão aplicadas três técnicas de recolha de dados:

- Questionário de caracterização para obter informação que caracterize os dois grupos de participantes que se pretendem analisar;
- Entrevista semiestruturada que permita obter dados para avaliar a usabilidade do protótipo depois de se ter efetuado a parte prática da experiência;
- Guião de observação para medir os erros e outros aspetos importantes para relacionar com a entrevista;

Para que se possam obter dados fiáveis sobre a usabilidade do protótipo, vai-se realizar uma entrevista do tipo semiestruturada com cada participante. Segundo Bernard, a entrevista do tipo semiestruturada é utilizada com a maior eficácia, quando não é possível marcar uma nova entrevista ou quando se pretende fazer várias entrevistas para recolher dados importantes (como citado em Cohen & Crabtree, 2006). Não obstante, este tipo de entrevista permite que surjam novas perguntas, fruto de algum tópico mencionado que se considere importante (Alan Bryman, 2012) e que não foram pensadas na redação desta. Tendo em conta que alguns dos entrevistados são colaboradores da Altice Labs – empresa criadora do produto MEO, onde se realizou o estágio para desenvolver a componente prática do projeto – deve-se preparar uma entrevista que permita observar como os participantes respondam às questões que foram preparadas previamente no questionário de caracterização e encoraje uma comunicação mais informal no decorrer da

experiência, que se prevê ser fulcral para entender como é que os entrevistados reagem antes e depois de se sentirem mais à vontade, através do uso da linguagem natural, para interagir com um sistema televisivo.

O guião de observação da experiência será um instrumento importante para a recolha de dados e para registar informação qualitativa previamente definida com base na opinião do observador, no decorrer da entrevista.

2.2.1. Questionário de Caracterização

O questionário de caracterização serve para assinalar a amostra de participantes com base na área de formação, funções que desempenha e níveis de conhecimento, utilização de sistemas conversacionais, entre outros, para que mais tarde se consiga segmentar os resultados em amostras de dois grupos: participantes com e sem experiência em interagir com sistemas por voz. Esta divisão deve-se ao facto que se prevê que as pessoas que nunca interagiram por voz poderão ter mais dificuldades do que aqueles que a utilizem no seu dia-a-dia. Não obstante, conceptualizou-se uma avaliação que poderá resolver esta diferença entre os participantes, ao explicar previamente como é que o sistema e as funcionalidades que se pretendem avaliar funcionam, para que possam iniciar a experiência com um nível de informação semelhante.

Para que seja possível interagir por voz por linguagem natural, o sistema precisa de ser treinado ao nível do NLU com múltiplas formas de pedir a mesma ação. Inicialmente pensou-se em fazer uma entrevista para se proceder ao levantamento de frases, mas face a constrangimentos de tempo, optou-se por treinar o sistema com as frases que foram discutidas e aprovadas em reunião com a Altice Labs. Assim que o sistema conversacional esteja treinado para o contexto específico que se pretende avaliar e devidamente integrado com a aplicação MEO, será feita uma avaliação com participantes de diferentes literacias digitais, de modo que se possa entender o contributo da ILN para interagir com conteúdos do MEO face à interação tradicional com o telecomando.

2.2.2. Guião de observação

Este guião servirá para registar os dados relativos a erros de interação e outras notas consideradas importantes para o decorrer da experiência, que são orientadores da observação e acerca dos entrevistados sobre a forma como interagem com o sistema, através da linguagem natural. As observações que forem recolhidas serão apresentadas segundo uma escala de fraco até forte, que

pretende avaliar a qualidade da interação do utilizador com o sistema, para cruzar a análise dos dados obtidos com a entrevista semiestruturada.

2.2.3. Guião da Entrevista

Neste artefacto, serão contempladas as linhas orientadoras da entrevista, para que os seus objetivos sejam satisfeitos, remetendo ao tema central, ainda que seja do interesse de quem entrevista recolher também outros dados que sejam fruto da interação do entrevistado e que possam vir a ser considerados importantes para as conclusões da dissertação.

2.3. Limitações

No decorrer do desenvolvimento da investigação, algumas limitações podem ter condicionado os resultados da mesma, nomeadamente o número reduzido de participantes (15 participantes) que avaliaram o protótipo, os quais com idades muito dispersas, não sendo representativos de uma faixa etária específica. Do total dos participantes somente um terço são do sexo feminino. A amostra foi angariada por conveniência e apesar que nem todos conhecerem e utilizarem habitualmente sistemas conversacionais e/ou assistentes virtuais, os resultados obtidos não podem ser generalizados e são válidos somente para este contexto específico de utilizadores.

Outra limitação relaciona-se com o contexto empresarial em que a investigação foi desenvolvida, mais concretamente a localização e o horário semanal do mesmo, que interferiu na disponibilidade de outros participantes e os protocolos de segurança que foram condicionantes da investigação.

3.Estrutura da dissertação

O trabalho em causa é apresentado por vários capítulos de modo a que o conteúdo se encontre mais estruturado e organizado, facilitando o acesso e a compreensão do mesmo.

Sendo assim, após este capítulo de **introdução**, que pretendeu efetuar uma breve descrição do trabalho, fornecendo uma contextualização do mesmo, da problemática do estudo e da metodologia que o mesmo irá seguir, o capítulo segundo, **enquadramento teórico-prático**, pretende explicar e descrever os principais conceitos que servem de base para a realização da investigação e do projeto.

No terceiro capítulo, **o desenvolvimento**, pretende explicar e descrever todas as decisões tomadas para a recolha de frases, nas fases do processo de implementação sobre o treinamento do protótipo na componente de *natural language understanding* e na etapa de integração da interação por linguagem natural, com o sistema televisivo MEO. No final do capítulo é descrito todo o processo realizado para a avaliação do protótipo, acompanhado com os respetivos resultados e propostas de soluções para os problemas identificados.

No quarto capítulo, **a Recolha de dados e discussão dos resultados**, são expostos os resultados obtidos com a realização da investigação e desenvolvimento.

Para finalizar, nas **conclusões**, serão apresentadas as limitações que o mesmo enfrentou, tendo sempre como base a questão de investigação e objetivos propostos inicialmente. Para além disso serão feitas algumas propostas de trabalhos futuros de melhoria.

II.Enquadramento teórico-prático

Este capítulo começa por introduzir o conceito de interação por voz na perspetiva da sua evolução histórica, desde o primeiro sistema de reconhecimento por voz conhecido até à atualidade, apresentando os avanços tecnológicos mais importantes que contribuíram para o seu desenvolvimento. Para compreender o estado de integração atual desta tecnologia nos nossos sistemas televisivos, fez-se um levantamento de soluções de mercado, destacando quais são as empresas consideradas as mais influentes. Para além disso fez-se um levantamento de soluções com interação por voz desenvolvidas no domínio académico/científico, para analisar as investigações e desenvolvimentos mais recentes, que convergem para a subcomponente mais sofisticada da interação por voz, conhecida como interação por linguagem natural.

4.Interação por voz

A experiência de utilizar comandos de voz para controlar computadores deu origem a uma nova geração de sistemas com este modo de interação. A interação por voz encontra-se num estado de evolução que se compara a uma criança que começa a aprender a andar, mas que cai regularmente e tem momentos em que perde o equilíbrio. No entanto, é possível perceber que, através da análise histórica, já percorreu um longo caminho, comprovado pelas várias ofertas de mercado que transformam a forma como os seres humanos interagem com objetos/aplicações, utilizando comandos por voz, ou comunicando com assistentes virtuais que são capazes de compreender o utilizador através da linguagem natural.

4.1. Evolução histórica da interação por voz

Desde que se inventou o primeiro computador mecânico, começou-se a desenvolver um fascínio por tentar falar com máquinas, como se estas fossem seres humanos. Este sentimento começa, na prática, a manifestar-se uma década após o desenvolvimento do primeiro computador digital completamente funcional, conhecido como ENIAC (Figura 1), em 1942 (Hope, 2017).

Figura 1 - Primeiro computador ENIAC

O primeiro sistema de reconhecimento por voz conhecido foi desenvolvido entre 1950 e 1960 e este período foi denominado por Baby Talk (Pinola, 2011), graças à evolução dos sistemas criados como a Audrey, que apenas era capaz de reconhecer dígitos e o sistema Shoebox da IBM, que já conseguia reconhecer 16 palavras de língua inglesa. Apesar destes acontecimentos criarem interesse na comunidade científica, só na década de 70 é que se apostou na investigação de sistemas de reconhecimento (SR) por voz, através do *Speech Recognition Program* (SRP), que foi responsável pelo desenvolvimento do protótipo Harpy que continha um léxico de 1011 palavras. Esta década também marcou o mercado, pelo que neste período a Bell Laboratórios introduziu um sistema ambicioso capaz de interpretar vozes de diferentes pessoas. A década de 80 foi a principal mobilizadora para o desenvolvimento de sistemas preditivos através do modelo estatístico conhecido como *Hidden MArkov Model,* que considerava a probabilidade de que os sons desconhecidos poderiam ser palavras (Juang & Rabiner, 2004). A evolução dos SR levou a novos produtos comerciais, como a boneca Jullie (Figura 2), que continha um vasto vocabulário destinado para interagir com crianças ("1980's Julie by Worlds of Wonder Commercial - YouTube," 2006).

Figura 2 - Julie a ler a sua história ao utilizador

No entanto, todos estes sistemas apresentavam problemas na forma como as palavras eram processadas, obrigando o utilizador a dizer palavra a palavra de forma pausada. Em 1990, graças à constante evolução da capacidade de processamento dos computadores, esta tecnologia é massificada para o público. Surgiram softwares comerciais como o SR Dragon[2] que era capaz de reconhecer o utilizador a falar naturalmente, com uma capacidade máxima de 100 palavras por minuto. Não obstante estes avanços, o problema do SR mantém-se, sendo que a precisão em compreender palavras destes sistemas não ultrapassava os 80% (Pinola, 2011), tornando-os pouco eficazes. A Google adicionou, em 2010, a funcionalidade de reconhecimento personalizado para que fosse possível gravar as pesquisas de voz dos seus utilizadores, com o objetivo de aumentar o grau de precisão do modelo. Em 2011, a Apple lançou o SR Siri que, tal como a tecnologia implementada pela Google, processa informação baseada em *Cloud* (Pinola, 2011). Com a evolução destes dois sistemas, a interação evoluiu do panorama típico de utilização de comandos por voz

[2] https://www.nuance.com/dragon.html

para a capacidade de responder a questões mais complexas graças às potencialidades de interação por linguagem natural (ILN), que tanto a Google *Voice Search*, como a Apple Siri apresentam, assim como outras soluções conhecidas no mercado como a Alexa e a Cortana, desenvolvidas respetivamente em 2014 pela Amazon (Weinberger, 2017) e pela Microsoft (Corden, 2017). Estes assistentes abrem novas possibilidades de interação e não apenas para executar as tarefas que foram pensadas inicialmente. Por exemplo, a Alexa foi desenvolvida inicialmente para ser uma assistente virtual com uma vertente associada à área da domótica, com um carater lúdico, como se pode observar na Figura 3, e noutra vertente,foi direcionada para a interação com sistemas televisivos por linguagem natural (Figura 4).

Figura 3 - Utilização lúdica do Amazon Echo

Com a rápida evolução do paradigma televisivo, passando da tradicional comutação de canais para uma vertente mais interativa onde os conteúdos são os elementos centrais, é cada vez mais importante refletir sobre como se interage com a televisão (TV). As aplicações de TV são cada vez mais complexas, com funcionalidades que são por vezes difíceis de utilizar com o tradicional telecomando. Através destas limitações, surge uma necessidade de adaptar como interagimos com o ecossistema televisivo, para um tipo de interação que permita ao utilizador interagir com o sistema de forma mais natural, através da voz (Figura 4).

Figura 4 - Exemplo de interação com um sistema televisivo

A interação por voz não é uma novidade na interação em aplicações, mas as suas potencialidades ainda não foram totalmente exploradas. Os tipos de interação por voz existentes no mercado permitem apenas a interação por comandos, como por exemplo aumentar o volume, mudar de canal, entre outros, recorrendo a frases simples e curtas, uma vez que só existe uma frase ou palavra que permite a execução de cada comando, o que leva a um maior esforço cognitivo do utilizador, ao ter de memorizar cada comando e a respetiva sequência de palavras. No entanto, com a evolução da interação por linguagem natural (ILN), é possível criar sistemas conversacionais Humano-Computador, que sejam capazes de criar e manter um discurso contínuo, refinando o resultado da pesquisa, mediante sucessivas interações do utilizador com um léxico de palavras mais abrangente. Desta forma, podemos executar pesquisas mais complexas, melhorar a qualidade da comunicação por voz e criar uma maior empatia com quem comunica com o sistema. Interagir com um sistema televisivo por voz natural, apresenta uma complexidade de implementação grande, visto que é necessário dotar o sistema de vocabulário que lhe permita entender as mais variadas formas de fazer um pedido sobre uma determinada funcionalidade. Esta complexidade é demonstrada através da tecnologia que foi e continua a ser adotada pelos grandes *Players*, que são os comandos de voz limitados, pela rápida implementação dos mesmos. Não obstante, já existem *Players* que integram nos seus sistemas interação de voz por linguagem natural.

4.2. Levantamento de soluções de mercado com interação por voz

Interagir naturalmente com um sistema televisivo por voz apresenta uma complexidade de implementação grande, visto que é necessário dotar o sistema de vocabulário que lhe permita entender as mais variadas formas de fazer o mesmo tipo de pedido sobre uma determinada funcionalidade. Segundo Kathryn do grupo Nielsen Norman (Whitenton, 2017), para atingir um nível de interação com sucesso através da voz, para além do sistema ter de ser muito bom na compreensão de linguagem natural, as interfaces de voz necessitam de possuir estratégias para

ajudar os utilizadores a perceberem as ações que podem tomar para atingir um objetivo e também perceber qual é o resultado derivado dessa interação. Na interação por voz, como não existem sinais visuais, os utilizadores necessitam de decorar comandos ou pensar como deverão interagir com o sistema. Desta forma, a interação precisa de informar ao utilizador das suas ações, seja através de significantes de sons, efeitos visuais, ou ambos.

> *Definition: A voice-interaction signifier is a user-interface cue that the system provides to users in order to help them understand what verbal commands they can make* (Whitenton, 2017).

Existem três tipos de significantes de interação por voz a considerar: não verbais, explicitamente verbais e implicitamente verbais. Os significantes não verbais são gerados pelo sistema e dão *feedback* sobre ações especificas, como por exemplo, a ativação de um diálogo, assim que percebe que o mesmo acabou, ou quando não entendeu corretamente o que o utilizador quis dizer. Os significantes explícitos são utilizados para confirmação de que o sistema entendeu o que o utilizador disse e para sugerir que opções estão disponíveis com base no que foi pedido, por exemplo, quando pedimos ao Google para marcar uma reunião, ele responde "Ok, para quando?". Os significantes implícitos servem para quando o sistema deteta a palavra de ativação e para o seu discurso, para permitir que o utilizador dê uma nova instrução, imitando assim, a forma como os humanos interagem em diálogo, parando o discurso para que o outro interveniente possa falar. O uso de forma equilibrada destes significantes de voz, torna o discurso mais percetível para o utilizador, visto que os seres humanos não têm capacidade para decorar um discurso completo. Os significantes de som explícitos podem servir para este efeito, solucionando ações que poderão ser irreversíveis, como o caso de uma compra online. No entanto, o uso constante deste significante pode tornar o diálogo aborrecido e extenso, pelo que deve ser apenas utilizado para a recapitulação de um comando extenso, ou de uma intenção. A habilidade que o sistema tem de adivinhar qual é a intenção do utilizador, contribui para interações mais eficientes se o sistema for treinado previamente com informação suficiente, como se pode observar na Tabela 1. Os quatro exemplos de diálogo dos sistemas representados fazem um uso equilibrado dos três tipos de significantes, embora que, através do exemplo da figura seja apenas possível visualizar dois, o significante implícito para a confirmação de entrada de dados e na introdução de uma nova questão e significantes explícitos para recapitular o pedido, caso seja necessário alterar algum dado. Não obstante, o sistema da Google e da Siri não são capazes de utilizar significantes para edição de detalhes de um evento, como o caso do Echo e da Cortana, e passam para uma pergunta explícita para guardar o evento. O problema desta interação é que as questões que ambos utilizam para este fim, "*Shall I schedule it*?" e "*Do you want to save this*?" são perguntas com resposta de sim e não e apesar de ser possível editar detalhes, o utilizador é que tem de perceber que existe a possibilidade de o fazer. Desta forma, a utilização de significantes implícitos para os cenários de confirmação é muito mais eficiente porque permitem reduzir o número de interações necessárias realizar uma ação.

Tabela 1 - Diferença entre respostas dos vários serviços de ILN (Whitenton, 2017)

Voice system responses to the user command "Set up a meeting at nine."			
Google	**Echo**	**Cortana**	**Siri**
What's the title of the event? Haircut **Sure, Haircut tomorrow at 9 am. Do you want to save this?** Yes **All right, I saved your event.**	**Is that nine o'clock in the morning or in the evening?** In the morning **On what day?** Thursday **And what's the name of the event?** Haircut **I'll schedule Haircut on Thursday, August 24th at 9 am, ok?** Ok **Ok, I've added that.**	**What do you want to call it?** Haircut **I'll add Haircut to your calendar for today at 9 pm. Sound good?** No, change it to 9 am tomorrow **Sorry, should I add this to your calendar?** No **Did you want to change the title or the time?** Time **When is your event?** Nine am tomorrow **All right, I'll add Haircut to your calendar for tomorrow at 9 am. Sound good?** Yes **Sure thing.**	**Ok, I set up your meeting for today at 9 pm. Shall I schedule it?** No, change it to 9 am tomorrow **To continue, you can confirm, cancel, change the time, or change the title.** Change the time **What time is your appointment?** 9 am tomorrow **Ok, I set up your meeting for tomorrow at 9 am. Shall I schedule it?** Change the title **Ok, what's the new name for this meeting?** Haircut **Ok, I set up your meeting for tomorrow. Shall I schedule it?** Yes **It's on your calendar for 9 am tomorrow.**

Os sistemas conversacionais devem ser concebidos com tempos de interação reduzidos para que se possa minimizar erros durante o diálogo. A complexidade existente para que os sistemas conversacionais tenham um bom funcionamento é demostrada através da tecnologia de interação com sistemas de televisão que foram e continuam a ser adotadas pelos grandes *Players*, que são os comandos de voz, uma vez que o desafio técnico é menor e de rápida implementação. Não obstante, também já existem soluções no mercado que integram a interação de voz por linguagem natural e que serão mencionadas ao longo do texto por ordem cronológica, quando no subcapítulo dos avanços tecnológicos de interação por voz. Os principais *Players* no mercado que possuem sistemas televisivos com suporte de interação por voz são a Google, a Apple e a Amazon, respetivamente com sistema Android TV, Apple TV, Amazon Fire TV. Para além da integração da voz em sistemas televisivos, a Amazon integrou o seu assistente inteligente no produto Amazon Echo e a empresa Harman Kardon incorporou o assistente inteligente Cortana da Microsoft num dos seus modelos de colunas mais recente, denominadas de Invoke.

Para esta análise do estado da arte, apenas se vão considerar sistemas de reconhecimento de voz desenvolvidos a partir de 2010 e que estão integrados no contexto televisivo. O primeiro sistema

de reconhecimento capaz de entender palavras foi o sistema de pesquisa por voz da Google em 2010, em sistemas Android. No início, este sistema contava com um léxico de 10 a 100 palavras e, atualmente, já conta com mais de 230 biliões de palavras (Pinola, 2011), reconhecidas por consultas (*queries*) de pesquisa de utilizadores reais (Katzmaier, 2017).
Mais tarde, em outubro de 2011 a Apple decidiu lançar um assistente inteligente conhecido como Siri, que é capaz de conversar naturalmente com o utilizador e que também funciona através de processamento de informação em *Cloud*, como os sistemas da Google e Amazon. A Siri consegue dar resposta aos pedidos com um tom de personalidade como se fosse Humana, através da concatenação de vários sons extraídos das gravações de palavras, que formam novas frases com sentidos diferentes (Zibreg, 2017). As respostas da Siri conseguem ser tão precisas e por vezes originais, que faz com que não seja apenas utilizada para realizar algumas tarefas do dia-a-dia, como também para servir de entretenimento, permitindo o dialogar sobre vários temas.
Em 2014, a Amazon apostou no sector de *Internet of Things* (IoT) lançando o Amazon Echo, com um assistente virtual pessoal denominado de Alexa (Amazon, 2018), que permite realizar tarefas de domótica, oferecer um conjunto de informação em tempo real a pedido, reproduzir música, entre outras funcionalidades. Este assistente virtual também foi integrado no Amazon Fire TV, para pesquisa de conteúdo por voz. Em 2015, a Apple TV incorporou a Siri (R. Williams, 2015), para pesquisa de conteúdos, por várias categorias, como título e autores, etc. através de comandos por linguagem natural limitados.

4.3. Levantamento de soluções de academia com interação por voz

Para se entender um pouco melhor a complexidade dos sistemas de ILN, de uma forma desarticulada das implementações comerciais nas quais há uma preocupação clara ao nível da oferta de um produto robusto, importa olhar para as investigações e desenvolvimentos que têm início na academia.

Como foi referido, na análise feita ao estado da arte da interação por voz, na perspetiva do mercado, depreende-se que a interação por voz, ao longo do tempo, tem evoluído até à forma como é atualmente conhecida, permitindo ser utilizada como um mecanismo de acesso de conteúdos a pedido. Esta interação também resultou da dificuldade em navegar nos atuais sistemas de televisão, que apresentam interfaces semelhantes às de um computador, com o uso de um tradicional telecomando (Berglund & Johansson, 2004). Apesar de, atualmente, as interfaces já adotarem um design *user centered* (Gorelick, 2017), que foca as necessidades e requisitos do utilizador (Norman & Draper, 1986) através da Usabilidade e também abordagens *content first* (Lafferty, 2016), a interação por voz aliada a estes tipos de abordagem de *User Interface* (UI) confere um acesso a conteúdos com maior acessibilidade, minimizando a taxa de erros e melhorando a experiência do utilizador. No entanto, apesar dos comandos por voz serem úteis e eficazes para a execução de comandos mais simples (mudar volume, canal…), têm algumas fragilidades inerentes a essa interação que a ILN procura resolver, como por exemplo, a necessidade de uma interação física para ativar o sistema de voz (que tipicamente se traduz em premir um botão com um logótipo de microfone) e o facto do utilizador ter de decorar as palavras chave para cada comando, de forma a conseguir ativar a respetiva funcionalidade. Com os avanços da interação de voz por linguagem natural, tornou-se possível a existência de sistemas conversacionais entre Homem-máquina (J.

Williams, Raux, Ramachandran, & Black, 2013), que solucionam os problemas mencionados nos comandos por voz que possuem um vocabulário limitado e só funcionam se for satisfeita a ordem frásica que foi programada, que são inerentes à interação por comandos de voz através de linguagem natural. Apesar que na ILN, o mapeamento de palavras é uma das fases cruciais para a criação de um bom sistema conversacional (Furnas, Landauer, Gomez, & Dumais, 1987), capaz de fornecer vários caminhos diferentes para aceder ao mesmo conteúdo, o léxico necessário para uma aplicação de televisão não tem a mesma abrangência e complexidade que um motor de busca como o da a Siri, uma vez que se enquadra num contexto mais fechado ao nível do número de interações exequíveis.

Só é possível devolver conteúdos sugeridos, se houver uma recolha de meta data sobre o consumo televisivo de um determinado utilizador. O sistema precisa de conhecer os hábitos de consumo de cada um dos seus utilizadores, com base em visualizações de conteúdo, hora do dia que é consumido e duração para ser capaz de conseguir oferecer potenciais conteúdos. Através de um sistema conversacional implementado por ILN é possível introduzir um diálogo para recolher informação de um conteúdo, como os autores, realizador, género, etc. Assim, o sistema consegue validar que informações são importantes para o utilizador, segmentando-as por níveis de preferência, evidenciando que atributos são importantes e melhorando futuras sugestões com más recomendações (Johansson, 2003). Quanto mais dados forem combinados, de uma forma analítica e através de mecanismos de personalização dentro da aplicação, maior é a probabilidade do conteúdo ser bem recomendado (Johansson, 2003). Para além dos dados gerados por um utilizador é possível sugerir novos conteúdos de uma forma eficaz, com base no consumo de outros utilizadores com perfis de consumo semelhantes, reduzindo a necessidade de implementação de um motor de sugestão muito complexo.

Estes mecanismos otimizados para a sugestão também são válidos para que um sistema conversacional possa interagir com o utilizador e ter a capacidade de perguntar e responder a um pedido de um utilizador, para voltar a ver um conteúdo, que não foi concluído numa sessão televisiva anterior. Essa ação de pedir não deve forçar o utilizador a aprender como deve interagir com o sistema. O sistema deve adaptar-se aos seus utilizadores e à forma como eles comunicam, criando vários caminhos diferentes que convergem para a mesma solução.

5.Interação por linguagem natural

A interação por linguagem natural permite que os utilizadores possam interagir com qualquer dispositivo da mesma forma como interagem com outros seres humanos. Isto é, esta tecnologia é inteligente o suficiente para perceber o significado das palavras e reagir de acordo, memorizando detalhes do diálogo e reagindo de acordo com o contexto, melhorando a experiência do utilizador. Para compreender como é que esta tecnologia é capaz de entrar em diálogo com este tipo de especificidades é necessário analisar a arquitetura do sistema e perceber quais são os componentes que a constituem, qual a contribuição de cada um para se gerar interação por linguagem natural e como é que esta pode ser ativada pelo utilizador.

5.1. Arquitetura do sistema de *Natural Language Processing*

Para compreender quais são as partes integrantes dos sistemas de *Natural Language Processing* (NLP), o diagrama da Figura 5 ilustra as várias etapas desde o momento em que o utilizador fala, até ao *feedback* do sistema.

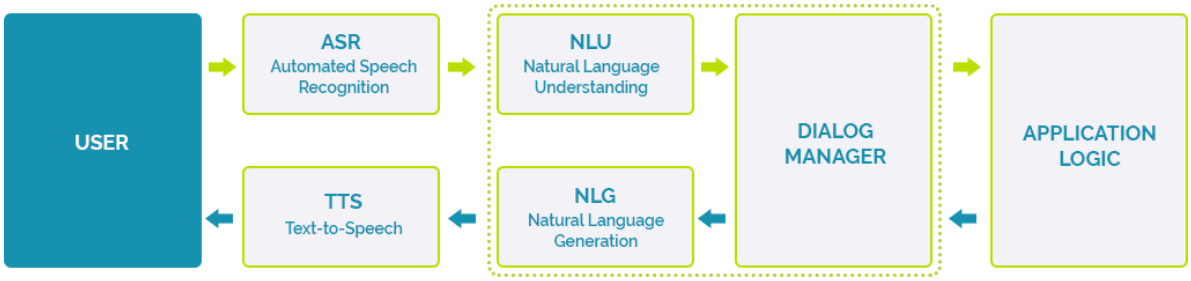


Figura 5 - Diagrama da Arquitetura de um sistema NLP (Slavetskiy, 2016)

O *Automated Speech Recognition* (ASR) pode ou não fazer parte da NLP, uma vez que é utilizada na conversão da comunicação por voz para texto. Os resultados obtidos pela ASR são enviados para a componente de *Natural Language Understanding* (NLU), sendo uma das partes integrantes do *Dialog Manager* (DM), responsável por atribuir sentido às palavras captadas pelo ASR e de seguida, o resultado que ainda não está materializado em linguagem natural, é recebido pelo Natural *Language Generation* (NLG) para ser convertido em linguagem corrente. Por fim, depois da voz ser convertida em texto e o DM lhe ter dado sentido (através das componentes de NLU e NLG), obtém-se um resultado textual, na forma de linguagem natural que o utilizador entenda e, por fim, o produto final é enviado para conversão para voz no motor text-*to*-*Speech* (TTS).

5.2. *Automated Speech Recognition*

Automated Speech Recognition é um componente da interação por voz, que permite reconhecer e traduzir voz para texto, através de um computador. Alguns destes sistemas de reconhecimento são dependentes e necessitam de ser treinados, havendo a necessidade de um indivíduo ler pequenos trechos de um texto ou palavras isoladas para o sistema, aumentando dessa forma a eficácia do reconhecimento. No entanto, também existem sistemas que não necessitam de treino e aprendem com base no diálogo com outros indivíduos, sendo denominados por reconhecimento independente (Zajechowski, 2014).

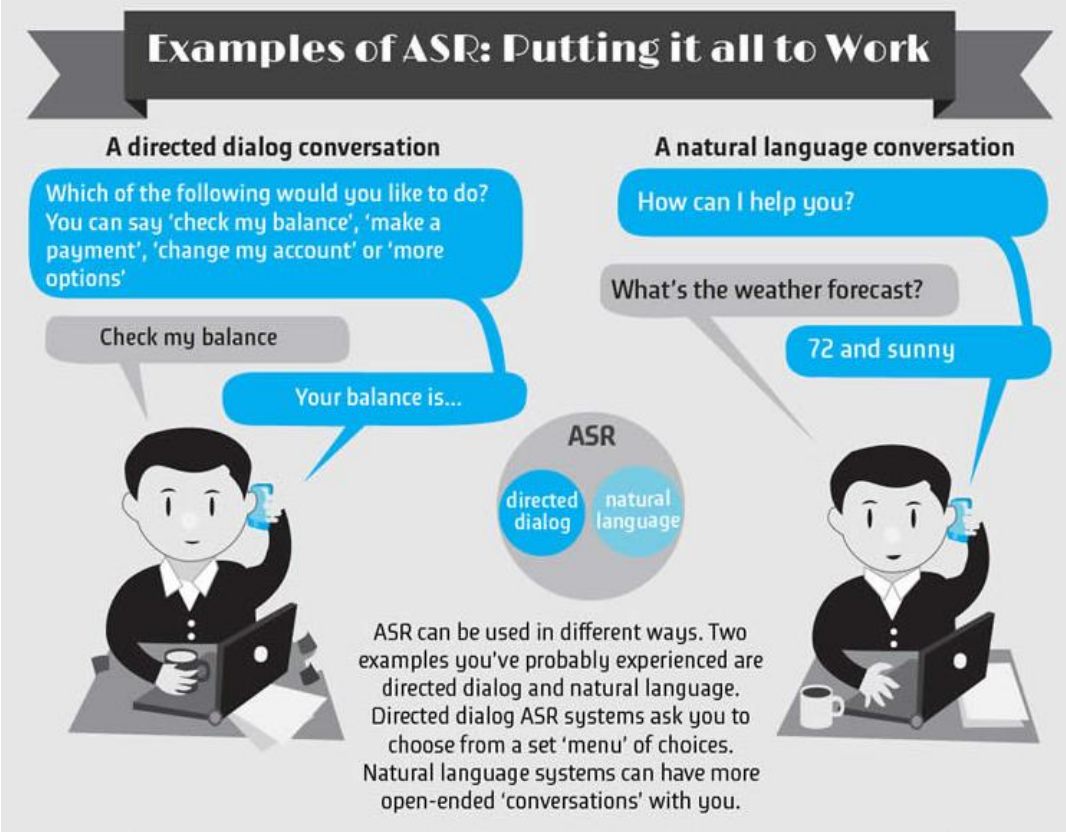


Figura 6 - Exemplo de ASR (Zajechowski, 2014)

Dos motores de ASR existentes, destaca-se o Bing Voice da Microsoft, o Google Cloud Platform, o IBM Watson e Nuance Dragon, que através da análise feita pela Altice Labs, foram definidos como sendo os componentes mais robustos de *speech to text* para desenvolvimento disponível no mercado. Um dos aspetos positivos destes motores de ASR é a possibilidade de poderem ser utilizados de forma independente das outras tecnologias que compõem o sistema de NLP. Este é um fator importante para a fase de implementação, uma vez que será possível utilizar os componentes que integram o NLP de diferentes proprietários, podendo-se optar por utilizar os componentes de cada parte do sistema que melhor se adequam a cada projeto.

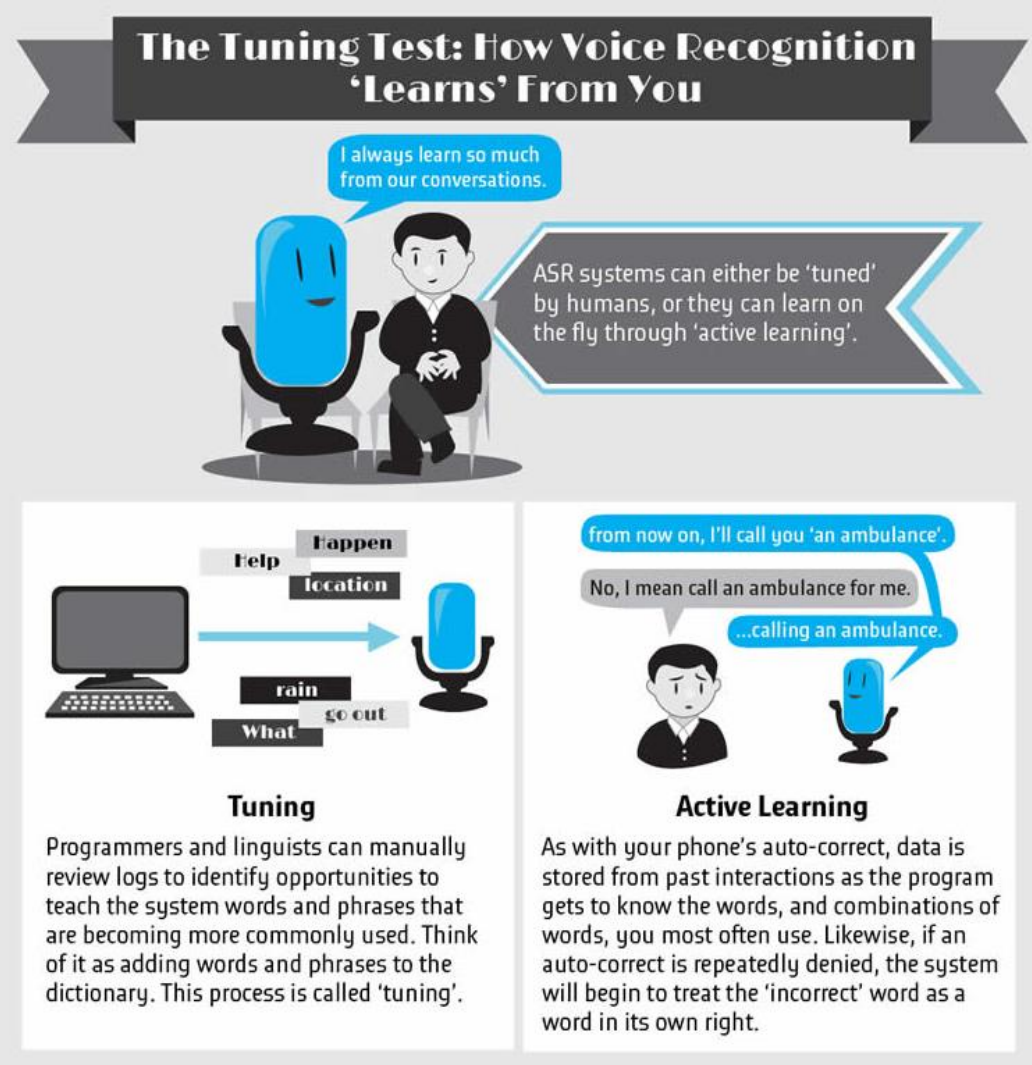


Figura 7 - Exemplo de tipos de ASR (Zajechowski, 2014)

A Microsoft desenvolveu o serviço Bing Voice (Microsoft, 2017), que é capaz de reconhecer áudio em tempo real através de um microfone ou através de um ficheiro de áudio. Como tal, após a informação ser enviada para o servidor, obtém-se um resultado parcial do reconhecimento de áudio com uma previsão do que o utilizador poderá estar a dizer sem ter acabado de falar ("Get started with the Microsoft Speech Recognition API by using the C# desktop library | Microsoft Docs," 2017). Os resultados desse reconhecimento são tipicamente apresentados ao utilizador como feedback gráfico e textual na aplicação sobre o que acabou de dizer.

Os serviços da Google Cloud platform (Google, 2017) convertem voz em texto, através de modelos de redes neuronais e é capaz de perceber mais de 110 idiomas e variantes diferentes, sendo, por isso, uma das API's mais completas do mercado. Para além disso, a sua robustez permite contornar o ruído exterior, conferindo uma melhor eficácia ao nível do reconhecimento de som possível. Também consegue reconhecer comandos com base num contexto específico, graças à personalização de palavras e frases, que são úteis para o vocabulário de cada aplicação e também

para o controlo de voz. Tal como o serviço Bing Voice da Microsoft, o Cloud Platform também retorna dados de reconhecimento enquanto o utilizador está a falar.

O serviço da IBM Watson (IBM, 2017) permite identificar e transcrever automaticamente o que está a ser dito por voz, ou num ficheiro de áudio e tem suporte para múltiplos formatos e várias línguas (Árabe, Inglês, Francês, Português do Brasil, Japonês e Mandarim). Para além disso, consegue entregar transcrições de grande fiabilidade, mesmo com ficheiros de áudio com baixa qualidade.

O Dragon (Nuance, 2017) é um serviço da Nuance, que para além das funcionalidades de reconhecimento e transcrição de voz, está integrado com ferramentas de escrita como o Microsoft Word aumentando a eficiência da escrita, através de uma comunicação híbrida entre comandos por voz e linguagem natural.

5.3. Análise semântica

Para se dar sentido na comunicação por voz mediada entre seres humanos, faz-se um uso inconsciente da entonação, do ritmo e da acentuação, de forma que a pessoa à qual se destina a mensagem, perceba as mudanças do tom de voz e adeque o seu discurso. Um exemplo prático é quando não se percebe a mensagem verbalizada e pede-se ao emissor para repetir, para que ele consiga explicar novamente com um tom mais calmo e audível, para que possa ser compreendido.

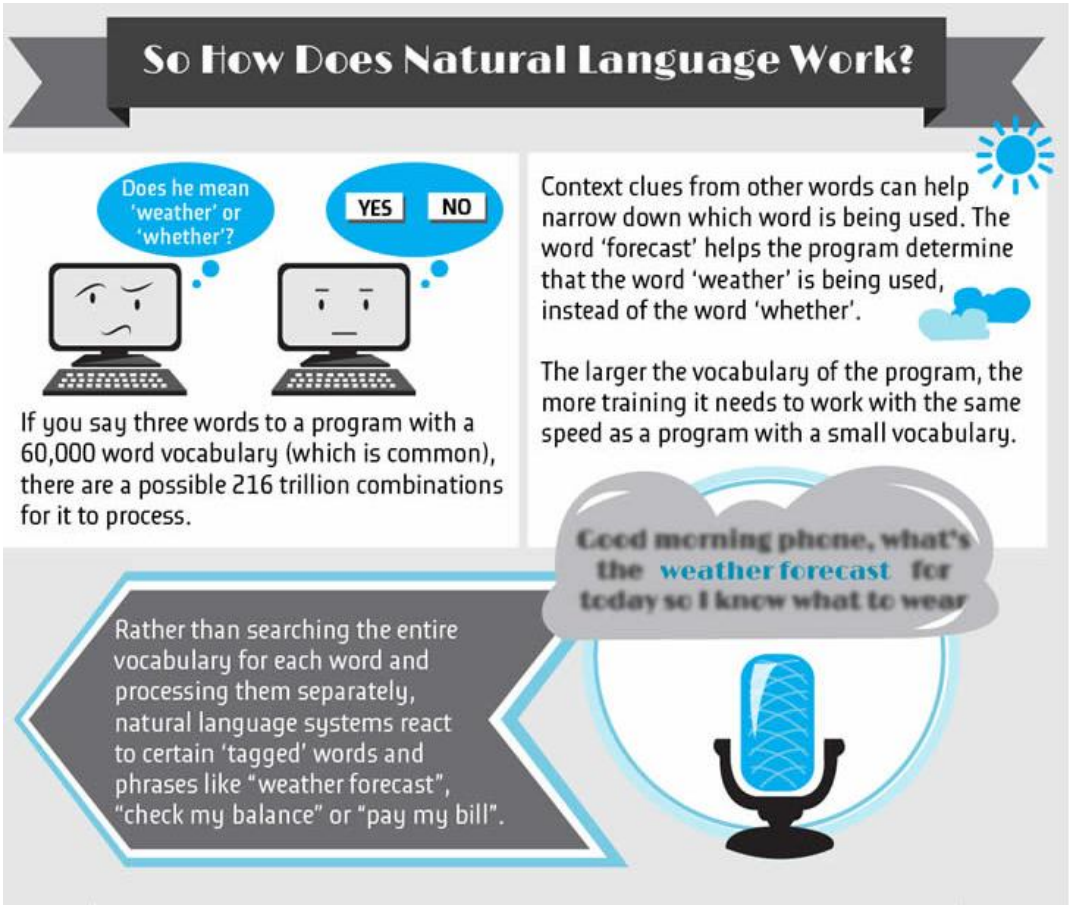


Figura 8 - Diferença entre comandos por voz e linguagem natural (Zajechowski, 2014)

5.3.1. Dialog Manager

O *Dialog Manager* tem um papel fundamental para a resolução deste problema. Este componente é um elemento constituinte da NLP e é responsável pelo estado e fluxo de uma conversa mediada entre um utilizador e um sistema conversacional. O DM subdivide-se em duas partes, *Natural Language Understanding* (NLU) e *Natural Language Generation* (NLG). Este componente é capaz de guardar variáveis de estado, como por exemplo, questões que ficaram por responder, por não ter sido treinado e um histórico da conversa, para que se seja possível melhorar o discurso, através do treinamento de léxico que ainda é desconhecido pelo sistema.

5.3.2. Natural Language Understanding

A Natural Language Understanding (NLU) é um subtema de uma área mais abrangente, conhecida como Natural Language Processing e são geralmente confundidas como se expressassem o mesmo (Healey, 2016). Depois de feita a conversão de áudio para texto, é necessário que o sistema entenda o que o utilizador quis dizer e defina o contexto, servindo-se de um conjunto de regras, modelos probabilísticos e outras técnicas que permitam afinar o discurso. Tipicamente estes componentes precisam de ser treinados e usam o léxico de uma linguagem e de regras de gramática para conseguirem segmentar uma frase para uma representação interna. A qualidade da NLU deriva de um bom vocabulário, teoria semântica e no caso da interação por linguagem natural, a inferência lógica é utilizada para deduzir conclusões (Covington, 1994; Gazdar & Mellish, 1989). Desta forma, a componente de NLU adaptada a um contexto melhora consoante o esforço que é feito para a treinar.

Quando recebe a voz na forma de texto – que é conhecido como *utterance* - da componente de ASR, a NLU classifica essas *utterances* através de *intents* e extrai o respetivo conteúdo, denominado de *entities*.

Os *Intents* são classes da utterance que captam a intenção do input, por exemplo, quando o utilizador se refere a tipos de filmes como comédia, ação, entre outros, para o NLU ele está a referir-se a um único *intent, "*Tipos de Filmes" com a respetiva informação (neste caso seria comédia, e ação) denominada de *entities*, como se pode observar no exemplo de input/output que a NLU recebe através da Figura 9.

```
{
    "tokenized_sentence": "utterance",
    "intents": [
        {
            "score": intent_probability_score,
            "intent": "intent_type"
        },
    ],
    "entities": [
        {
            "text": "text",
            "end": endpoint,
            "start": startpoint,
            "entity": "entity_type"
        },
    ],
    "sentence": "utterance"
}
```

Figura 9 - Exemplo de Input/Output da NLU (KITT.AI NLU, 2018)

As tecnologias de NLU que vão ser apresentadas são o LUIS e o Watson da IBM, fruto da mesma análise feita pela Altice Labs, sendo passíveis de serem utilizadas de forma independente das restantes componentes de NLP.

O *Language Understanding Intelligent Service* (LUIS) consegue perceber o que é que esses comandos dos cognitive services da Microsoft significam, para que o sistema consiga responder com certeza, às perguntas do utilizador (Barraza & Zhang, 2016).

O sistema Watson é capaz de reconhecer as necessidades dos seus utilizadores e responder com mensagem personalizadas, adaptando-se com a experiência. Para além disso, consegue perceber nuances e descobrir o contexto do que foi dito de forma intuitiva, fazer recomendações dirigidas a públicos específicos e encontrar novas oportunidades de descoberta, através da combinação de várias fontes de informação, extraindo a informação mais importante, sendo capaz de antecipar problemas antes que estes aconteçam (“IBM Watson,” 2017).

5.3.3. Natural Language Generation

Para converter os resultados da NLU em linguagem natural, o Natural Language Generation (NLG) funciona como um tradutor, processando a informação semântica que recebe para uma representação de linguagem natural. A sua função principal é a de tomar decisões sobre que palavras deve utilizar para dar o significado certo e formar um discurso que seja naturalmente percetível para o utilizador. Esta função subdivide-se em seis etapas, determinação de conteúdo, estruturação do documento, agregação, escolha lexical, geração de referências expressivas e a realização.

Quando a componente de NLU gera uma representação semântica é necessário determinar que conteúdo é importante de mencionar e estruturá-lo para obter a melhor organização de informação. Uma vez que podem surgir frases com o mesmo significado, ou que se podem relacionar, é necessário agregá-las, proporcionando uma interação mais natural e tornando-as de fácil leitura. Depois destas grandes alterações estarem bem definidas, é necessário refinar o léxico que melhor satisfaz o significado do que se pretende. De seguida, faz-se uma escolha de quais os pronomes e outros tipos de anáforas que vão ser utilizados e, por fim, cria-se o texto e revê-se questões de sintaxe, morfologia e ortografia.

Uma das grandes limitações na interação por voz, relativamente à conversão de texto em voz que advém da interação por linguagem natural, é a prosódia, principalmente em questões ligadas com a entoação, que faz com que o som gerado tenha sempre o mesmo nível de intensidade, afetando a forma como estes motores de *Text-to-speech (TTS)* soam.

5.4. Motores texto-to-speech

O processo de TTS é a última fase que compõe a interação por voz e traduz-se na habilidade que uma máquina tem de converter texto em voz, com uma panóplia de línguas, dialetos e vocabulário especializado por terceiros. Existem muitos serviços de TTS *open source* e outros fornecidos como ferramentas de desenvolvimento para aplicações. Para este estudo, apesar da vasta oferta de motores TTS, como os *screen readers*, apenas se vão analisar os motores de TTS que são utilizados para o desenvolvimento de aplicações.

Dos serviços TTS open source, destacam-se o Festival Speech Synthesis System, o eSpeakNG e o Gnuspeech, por serem atualizados e melhorados com regularidade e por pertencerem a organizações sem fins lucrativos e Universidades. O Festival Speech Synthesis System funciona através de uma síntese baseada em *diphone*[3]. O *eSpeakNG* possuí um uma sonoridade técnica melhor do que o *Festival System* e também tem suporte para mais línguas. A voz produzida pelo motor Gnuspeech assemelha-se à de um Humano, através de fala artificial, com base em regras de discurso articulado e modelos de ritmo e entonação.

Os TTS para desenvolvimento de aplicações mais conhecidos são o Google Deep WaveNet, Amazon Polly, Microsoft Bing Speech e IBM Watson, fornecidos através de API's. Estes motores apresentam uma voz natural, semelhante à de um Humano, no entanto as especificidades de cada um variam, havendo algumas limitações, como por exemplo, pronunciar incorretamente algumas palavras, como a palavra inglesa *lead*, que tanto pode significar chumbo ou pista e que se leem de forma diferente, apesar de se escrever da mesma forma (Lemmetty, 2017).

O motor de TTS da Google Deep WaveNet reconhece Inglês e Mandarim e processa as palavras através de modelos de redes neurais avançados, fornecido através de uma API. O *output* resultante

[3] Unidade de fala composta por dois sons de fala simples, conhecidos como fonemas

deste motor, apresenta sons de respiração e também, movimentos de boca, conferindo grande flexibilidade e características comuns na comunicação Humana.

O motor Polly da Amazon consegue compreender até 7 idiomas (inglês, dinamarquês, português de Portugal, português do Brasil, espanhol, japonês e coreano) e consegue reproduzir texto em linguagem natural, com uma grande variedade de vozes masculinas e femininas. Os tempos de resposta são muito rápidos e o preço desta API aumenta conforme a utilização. Apesar do motor TTS da Polly entender Português do Brasil, interessa referir que não está preparada para ser incorporado com o português de Portugal.

A Microsoft apresenta também, algumas funcionalidades como a possibilidade de alterar o dialeto de vários idiomas e vozes, modificar o volume e o pitch. A conversão TTS é rápida e está assente numa API REST.

A IBM desenvolveu o serviço Watson, que oferece a conversão em 7 idiomas (árabe, inglês, francês, português do Brasil, japonês e mandarim.) em tempo real, em vários formatos de áudio e interfaces de desenvolvimento. Este serviço permite guardar palavras chave, que sejam importantes e, também, possui a capacidade de distinguir diferentes utilizadores para não deturpar a qualidade do áudio (IBM, 2017).

5.5. Comandos

Para além dos componentes de TTS e NLU que a Altice Labs analisou, também foi objeto de estudo a forma como os comandos por voz podem ser ativados. Desta forma, foram identificados os seguintes dispositivos: telecomando Bluetooth com microfone operado por uma tecla, speaker com tecnologia microfone “far-field” (operado por uma wake up word), bem como um microfone “far-field” instalado na set-top box e um telemóvel.”

III.Desenvolvimento

1. Use case

Para validar a prova de conceito, foi necessário escolher um contexto específico do sistema televisivo MEO, para que o sistema conversacional, do qual a base é a interação por linguagem natural, fosse robusto e que permitisse explorar um diálogo com o sistema de forma extensa. Desta forma, o *use case* que se pretende analisar recaiu sobre como deve ser o diálogo resultante da interação por voz entre o sistema e o utilizador, possibilitando várias maneiras de pedir para ver um conteúdo que, por variadas razões, não foi visto até ao fim. Para alem disso, importa perceber que informação visual deve ser apresentada ao utilizador à medida que o discurso evoluí. Pretende-se ainda treinar a componente de NLU para um protótipo de um sistema conversacional, com a finalidade do utilizador poder, por exemplo, pedir para seguir uma série e continuar a ver um conteúdo mais tarde através da interação por linguagem natural.

Para que seja possível interagir com este sistema, vai-se desenvolver uma camada de interface em cima da aplicação do MEO, de forma a que quando se executar os pedidos para aceder a conteúdos sugeridos ou do tipo continuar a ver, o utilizador obtenha feedback visual, sobre o comando que utilizou e o respetivo resultado da interação por voz. O conceito de continuar a ver não é novo para a aplicação MEO, mas não tem nenhum tipo de inteligência associado à forma como categoriza os conteúdos para pertencerem ao tipo “continuar a ver”. Sempre que um utilizador começa a ver um conteúdo das gravações automáticas, o sistema guarda de forma automática o último instante do conteúdo que estava a ser visualizado. O valor que se pretende acrescentar a esta funcionalidade é o “continuar a ver” tradicional mais inteligente, permitindo separar os conteúdos que o utilizador escolhe guardar para continuar a ver e aqueles que são guardados automaticamente pelo sistema. Assim, é possível priorizar todos os conteúdos do tipo “continuar a ver” que resultarem das interações proactivas do utilizador, face aos conteúdos que são gravados automaticamente pelo sistema. Idealmente, foram pensados três cenários de utilização apresentados na Figura 10, Figura 11 e Figura 12, que em conjunto com a interação por linguagem natural, conferem uma experiência alargada para a visualização de conteúdos do tipo “continuar a ver” e que são abordados ao longo deste subcapítulo.

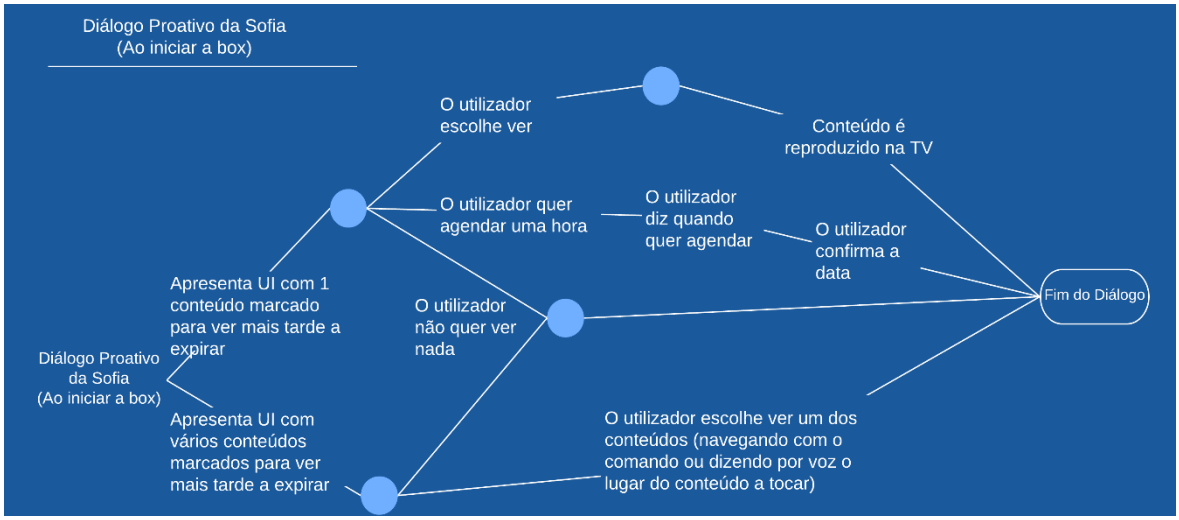


Figura 10 - Cenários de use case do protótipo - Diálogo da Sofia ao ligar a *set-top-box*

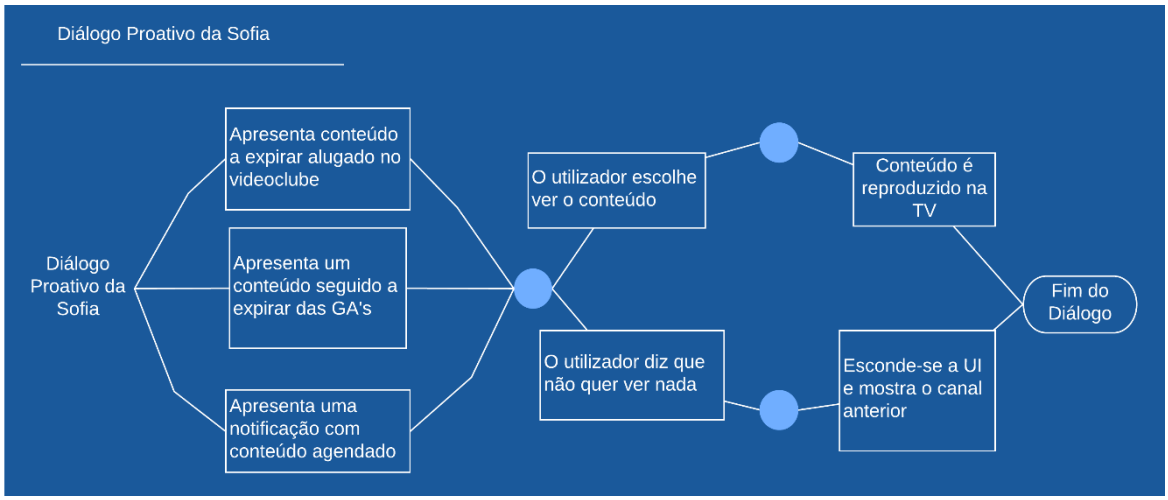


Figura 11 - Cenários de use case do protótipo - Diálogo proativo da Sofia

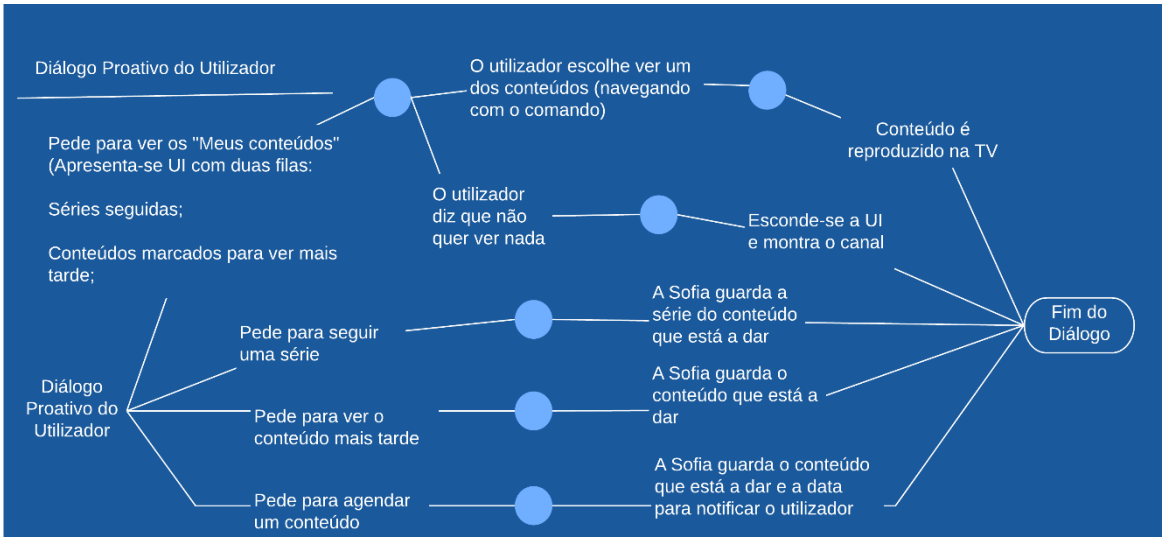


Figura 12 – Cenários de use case do protótipo - Diálogo proativo do Utilizador

Para que seja possível o utilizador pedir para continuar a ver um conteúdo é necessário analisar o léxico do universo televisivo e definir as formas mais recorrentes que os utilizadores poderão utilizar para pedir um conteúdo. Para efeitos deste protótipo (que será referido nos *use cases* como Sofia) pretende-se selecionar três exemplos que melhor caracterizam a interação e representem estes cenários. Quando o utilizador pede para visualizar um conteúdo mais tarde, o sistema deverá perguntar em que momento deve lembrar o utilizador, pelo que este deverá dar uma resposta sobre o momento desejado, ou eventualmente uma resposta menos conclusiva, como por exemplo, apenas "Mais tarde" e o sistema deverá lembrar o utilizador da próxima vez que este ligar o sistema. Quando esse momento chegar, deverá ser perguntado ao utilizador se deseja visualizar o conteúdo, pelo que terá várias formas diferentes de aceitar ou de rejeitar, dependendo do contexto do diálogo. O sistema deve interpretar o feedback e agir em conformidade com o pedido.

Idealmente, para seguir um conteúdo, o utilizador a qualquer momento poderá verbalizar para seguir um conteúdo e o sistema deve reconhecer o conteúdo que esta a dar no momento e executar o comando. Isto permite que a forma como são dispostos os conteúdos "continuar a ver" sejam dispostos por ordem de prioridade, consoante o que o utilizador pede para marcar. Apesar desta funcionalidade incidir mais nos comandos por voz do que num sistema conversacional, é importante para o sistema recolher mais informação sobre os hábitos de consumo do utilizador, para possa ser mais proativo. Nesse sentido, é possível criar vários momentos de diálogo com o utilizador, como por exemplo, o sistema recomendar novos episódios de uma série que o utilizador segue. Este comando também é importante para que o sistema consiga priorizar o conteúdo apresentado e, caso não existam conteúdos com a categoria "continuar a ver", possa recomendar conteúdos relacionados.

2.User Stories

Para que fosse possível recolher frases sobre como os utilizadores gostariam de interagir com a televisão para pedir conteúdos do tipo "continuar a ver", foram elaboradas três *user stories* que descrevem algumas situações em que os clientes MEO poderiam beneficiar se utilizassem um sistema inteligente que opere de forma proativa. A necessidade de criar cenários de utilização surgiu para apresentar as potencialidades da interação por voz, aliada às funcionalidades de "continuar a ver" e, também, para contextualizar o participante sobre que dados se pretendem recolher e quais os contextos específicos de utilização que se querem analisar, para implementar no protótipo. O objetivo desta recolha é, analisar o léxico televisivo que os participantes verbalizam para interagir com o sistema por linguagem natural, qual é a janela temporal recomendada para o sistema interagir com o utilizador e que possíveis *dead ends* possam existir no diálogo. Através desta análise, as *user stories* vão contribuir para se conseguir implementar um sistema que, através do treinamento da NLU, é capaz de interpretar o maior número de interações diferentes, dependentes do contexto de uso e que foram validadas por utilizadores com níveis diferentes de experiência em ILN.

2.1. Conteúdo de Videoclube a expirar

O Ivo costuma alugar filmes no videoclube, mas nem sempre consegue visualizar o que aluga até ao fim. Como costuma chegar a casa cansado, acaba por adormecer a ver televisão e/ou esquece-se que tem um tempo limitado para visualizar o conteúdo, obrigando-o a ter de o alugar novamente. A Sofia consegue saber quando um conteúdo está para expirar e pode lembrar o Ivo, para que não tenha de alugar novamente mais nenhum conteúdo que não tenha acabado de ver, conferindo uma melhor qualidade de experiência na televisão.

***Use cases*:**

- Como utilizador, quero ser lembrado pelo sistema sobre um conteúdo que aluguei no videoclube que está a expirar, para que o possa ver antes que acabe o tempo.
- Como utilizador, quero que sistema me sugira continuar a ver um conteúdo do videoclube, para que o possa ver em tempo útil.
- Como utilizador, quero que o sistema me sugira ver um conteúdo do videoclube que aluguei e ainda não visualizei.

2.2. Conteúdos de 7 dias a expirar

O Ivo gosta de assistir a todos os episódios das suas séries favoritas, recorrendo muitas vezes às gravações automáticas para os visualizar, conseguindo acompanha-los à medida que ficam disponíveis nas gravações automáticas. No entanto, quando eventualmente perde um episódio, sente que lhe falta uma peça importante para o seguimento da história. Para minimizar o risco de perder outro episódio, o Ivo pode pedir à Sofia para seguir uma série e ela o notificará se existir um conteúdo que esteja para sair das gravações automáticas. A Sofia também pode interagir com o Ivo de forma proactiva, notificando quando é que o próximo episódio vai estar disponível, para que possa agendar o momento em que ele deseja ver o episódio.

***Use cases*:**

- Como utilizador, quero seguir um conteúdo, para que possa receber notificações.
- Como utilizador, quero saber quando um conteúdo que sigo tem um episódio que não vi para sair das Gravações Automáticas.
- Como utilizador, quero saber quando um conteúdo que sigo tem um episódio que vai entrar nas Gravações Automáticas.
- Como utilizador, quero que o sistema me notifique de forma proactiva sobre o estado dos conteúdos que sigo.

2.3. Sugerir conteúdos quando se deteta que o utilizador adormeceu;

Depois do Ivo chegar a casa no fim do dia e fazer as suas tarefas diárias, chega o momento de descansar e de ver os seus conteúdos favoritos na televisão. Com o cansaço que acumula no dia-a-dia, por vezes torna-se difícil não adormecer, perdendo uma parte do episódio. Se a Sofia conseguir detetar que o Ivo adormeceu quando estava a ver televisão, pode lhe perguntar mais tarde se não viu o conteúdo até ao fim e sugerir-lhe continuar a ver esse episódio. Se no momento que a Sofia fizer a sugestão não for o mais oportuno, o Ivo poderá agendar para ver mais tarde e a Sofia o notificará nessa altura.

***Use cases*:**

- Como utilizador, quero que o sistema me sugira um conteúdo que não acabei de ver porque adormeci.
- Como utilizador, quero agendar quando quero ver o conteúdo, caso não consiga ver no momento em que o sistema interage comigo.

3.Levantamento das Tecnologias de ILN

As tecnologias escolhidas para o protótipo partem da experiência da empresa Altice Labs, fruto da análise preliminar de soluções de ILN. Das soluções analisadas, destaca-se as subcomponente de ASR – Automatic Speech Recognition (Zajechowski, 2014) e a componente de NLU – Natural Language Understanding (Rabiner & Juang, 2008) no âmbito do projeto de investigação Cooperative Holistic View on Internet and Content (CHIC). Este projeto reúne um consórcio de vinte e quatro instituições e tem como objetivo desenvolver plataformas digitais, em formatos abertos e tecnologias interoperáveis, bem como a promoção da dinamização de criação de conteúdos nacionais.

Uma vez que este projeto de dissertação faz parte do programa de bolsas GENIUS, que proporciona a realização desta investigação científica e desenvolvimento tecnológico para a obtenção do grau académico conferido pela Universidade de Aveiro em contexto empresarial, conta com uma parceria entre a Altice Labs e do grupo Social iTV da Unidade de Investigação DigiMedia (ambos membros no consórcio do projeto CHIC) do Departamento de Comunicação e Arte da Universidade de Aveiro. Não obstante, fez-se um resumo sobre as tecnologias mais atuais de cada Subcomponente, na temática de ILN, descrevendo por tecnologia sobre as soluções *cloud based* consideradas pertinentes, incluindo as que já foram identificadas pela Altice Labs até ao momento.

Destacaram-se, da análise feita das componentes de ASR e NLU, as tecnologias de *Cognitive services* da Microsoft (subcomponentes Bing Voice e LUIS), da Amazon (subcomponentes Alexa e Polly), da Google (subcomponentes Cloud Speech API e DialogFlow), da Nuance e Watson.

As soluções ponderadas pela Altice Labs, para a implementação do protótipo são, respetivamente, ao nível do ASR (*speech to text*) e da NLU, o Cloud Speech da Google e a Microsoft LUIS. Para a conversão de *text to speech,* fez-se um estudo constante no estado da arte, tendo sido selecionada a subcomponente da Amazon Polly pelo suporte à língua portuguesa de Portugal.

4. Arquitetura

O processamento da interação por voz, que ultimamente se traduz em comandos para executar na televisão com STB Meo, como evidenciado na Figura 13**Erro! A origem da referência não foi encontrada.**, é feito na área denominada de *Hands Free Device* com recurso a um Raspberry Pi. Este microcomputador contém a aplicação Sofia que corre numa camada por cima da *Framework* de JavaScript Node, responsável por moderar o fluxo da interação por voz, desde o momento em que o utilizador diz a *wake word* "Sofia" até à obtenção do comando a enviar para a aplicação Mediaroom, para apresentar a respetiva interface na televisão. Após a *wake word* ser proferida, o

que for dito pelo utilizador de seguida é guardado num ficheiro de áudio, que é enviado para a Google Speech API para o converter em texto. De seguida, é feito um pedido com o texto obtido para o Microsoft Bot Framework, que gere o estado atual do diálogo e, com base numa integração do NLU LUIS, este consegue perceber o que o utilizador está a dizer e atribui a respetiva intenção (*intent)*. O resultado obtido é depois enviado para a Web API Companion (também alojada no Raspberry Pi) que transforma o *intent* obtido pelo LUIS em comandos que a aplicação Mediaroom PF[4] executa na Televisão. As interfaces que são geradas, fruto de eventos gerados na aplicação Mediaroom, são alimentadas pela Web API[5] Proactive que recolhe e processa informação sobre a programação que está disponível para o utilizador consumir dentro do ecossistema MEO. Esta API recorre à base de dados baseada em *cloud*, Firebase Database[6], para guardar os conteúdos do tipo "continuar a ver", "Seguir", "Agendamentos" e dados de uma conversação do Bot Framework necessários para o sistema iniciar uma conversação proactiva com o utilizador.

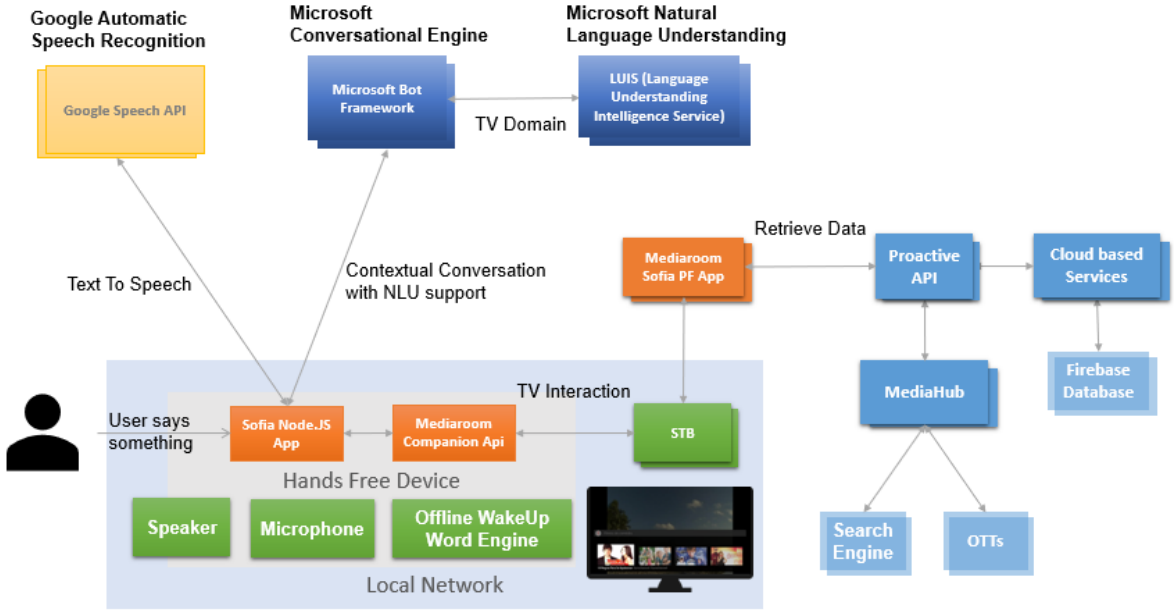


Figura 13 - Arquitetura do sistema

Para além das interfaces que dão resposta às interações do utilizador, os dispositivos de *input* e *output* de som, respetivamente o microfone e as colunas de som, também estão integradas no Raspberry Pi, tornando-o o um módulo indispensável para haver interação por voz. A API

[4] *Mediaroom Presentation Framework: http://www.mediakind.com/mediaroom.html*
[5] *Application Programming Interface*
[6] Link para o site do Firebase Database: https://goo.gl/T5rbDu

Companion alojada no Raspberry Pi comunica com a *set-top-box*[7] (STB) Meo através de uma rede LAN[8].

5.Implementação do protótipo Sofia

O nome sofia foi carinhosamente apelidado neste projeto, fruto de uma publicidade do MEO que estreou em 2018 com o humanoide Sophia. Este nome representa os avanços tecnológicos de interação por voz ao longo dos anos e que rapidamente ganha tração, podendo ser um fator impulsionante para o futuro da interação com a televisão.

O protótipo Sofia foi desenvolvido para dar resposta à questão de investigação proposta nesta dissertação, através da prototipagem uma solução de alta fidelidade, que reúne todos os requisitos necessários para tal.

O processo de prototipagem tem como objetivo, a conceção de fragmentos de um sistema, através do desenvolvimento rápido de interfaces e funcionalidades, de forma iterativa, com o objetivo de avaliar resultados obtidos com o protótipo (Maria Alves de Oliveira et al., 2007). Pretende-se avaliar as funcionalidades e interfaces propostas com o utilizador, para diminuir o esforço cognitivo e melhorar facilidade no acesso a conteúdos desejados, auxiliando-o através da realização de tarefas. Uma vez que a avaliação consiste em testes de usabilidade, para detetar problemas na interação quando o utilizador interage com o protótipo e aprender com o seu comportamento.

Uma vez que este projeto foi desenvolvido em contexto empresarial, utilizou-se como base um protótipo da Altice Labs, que permite interagir por voz através de linguagem natural, para que o foco deste projeto se centrasse na construção de frases e diálogos para o treino do NLU para a ativação das funcionalidades que se pretendiam implementar. A arquitetura do protótipo da Altice Labs é semelhante ao da Figura 13, com a exceção da inexistência dos serviços em *cloud*, sendo integrados para armazenar a informação dos programas, que provêm da interação com as novas funcionalidades desenvolvidas para esta dissertação. Não obstante, como cada aplicação serve um propósito específico, foi necessário implementar código em todas, para que as novas funcionalidades estivessem disponíveis e também, para se apresentar as novas interfaces que foram desenvolvidas especificamente no contexto desta dissertação. Para levar a cabo este desenvolvimento, foi necessário compreender o fluxo de informação do protótipo da Altice Labs e adquirir conhecimentos para a implementação, tanto ao nível dos componentes de voz, como também para o desenvolvimento de interfaces em Mediaroom PF. O protótipo e o contributo desta

[7] Dispositivo com um sintonizador de televisão, que recebe uma fonte de sinal e transforma em conteúdos para apresentar na televisão.
[8] *Local area network* é uma rede Ethernet ou Wi-Fi, que conecta vários dispositivos através de uma área limitada.

dissertação será explicado com mais detalhe ao longo deste capítulo de Desenvolvimento, que se encontra segmentado pelas várias aplicações que compõem o sistema.

Depois de se ter definido os componentes de interação por voz a utilizar neste projeto para o ASR, NLU e TTS, bem como as *user stories* que apresentam as funcionalidades que se pretendem implementar através de cenários de utilização, verificou-se uma necessidade de compreender como é que a informação que provém do *backend* do MEO está organizada e que tipo de dados a aplicação em Mediaroom PF Sofia necessita para reproduzir conteúdos audiovisuais do catálogo MEO na televisão. Deste modo, para entrar no fluxo da arquitetura que a Altice Labs já tinha desenvolvido previamente, começou-se por desenvolver um dos módulos para este protótipo, a *Proactive* API (*Application Programming Interface*).

As API's são conhecidas por fornecer um ponto de acesso entre aplicações e clientes, sejam eles utilizadores ou outras aplicações, permitindo uma interoperabilidade entre aplicações. Para este protótipo, dada a necessidade de integrar diferentes componentes para desenvolver uma solução, as API's funcionam como intermediários entre o sistema *hands free* e a aplicação Mediaroom, sendo atribuídos objetivos específicos para cada uma delas, cujas funções são mencionadas ao longo deste capítulo.

5.1. Proactive API

Esta API serve de mediador entre o sistema televisivo (cliente) e os serviços de informação provenientes do MEO. Como o *backend* do Meo não tem *endpoints* para sinalizar manualmente os conteúdos, quando o conteúdo fica a meio de ser visualizado (sendo guardado automaticamente pelo sistema), e de apesar de existir uma área de detalhe para cada conteúdo onde estão dispostos todos os episódios disponíveis, não é possível marcar uma série para receber as notificações de quando existem novos conteúdos ou quando estão para expirar das gravações automáticas. Para resolver esta limitação, fez-se uma interligação com uma base de dados Firebase Database, para guardar dados de conteúdos do tipo "continuar a ver", "seguir" ou que o utilizador agendou para ver mais tarde.

A aplicação Proactive tem um total de vinte e um *endpoints*. Desses, cinco são para guardar informação na base de dados relativos a conteúdos que o utilizador segue, conteúdos agendados, conteúdos marcados para ver mais tarde, por Id do programa (no caso de conteúdos disponíveis nas gravações automáticas), por título (no caso dos conteúdos do videoclube MEO), e para guardar as informações necessárias para que o Bot possa falar proactivamente com o utilizador. Dos restantes *endpoints*, sete são para comunicar com os serviços de *backend* do MEO, para obter informação sobre a programação, a disponibilidade de conteúdos nas gravações automáticas e os conteúdos alugados no Videoclube. Os restantes, organizam e processam a informação para alimentar as interfaces desenvolvidas, enviando os dados que são necessários para cada página implementada na aplicação Sofia Mediaroom PF.

5.1.1. Fontes de Informação

Numa primeira fase do desenvolvimento, fez-se uma análise de fontes de informação disponíveis no serviço MEO tendo em consideração o menor número de pedidos necessários para recolher toda a informação. Como se pode verificar na Figura 14, os serviços utilizados para a consulta à programação do MEO são os serviços de EPG (Electonic Program Guide), DVR (Digital Video Recorder), continuar a ver automático do MEO, o Search Engine e os conteúdos subscritos do Videoclube.

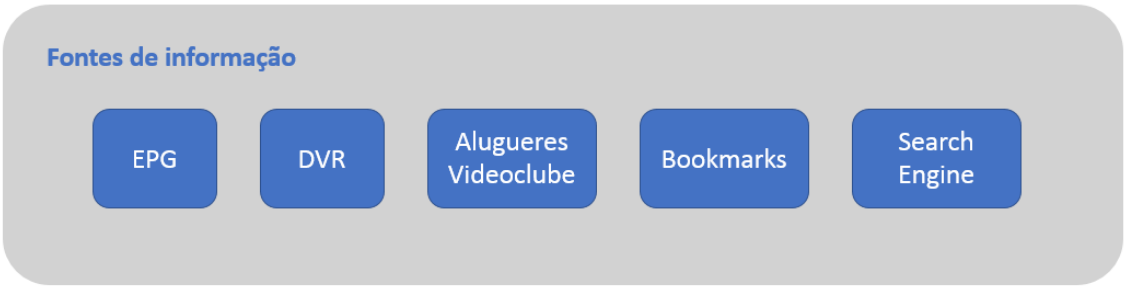


Figura 14 - Fontes de Informação do MEO

O serviço de EPG é utilizado para recolher informação do conteúdo linear que está a dar na televisão. Este serviço é invocado sempre que for necessário guardar um conteúdo na base de dados ou confirmar se este conteúdo existe no catálogo MEO. Quando um conteúdo não está disponível neste serviço, deixou de existir ou está disponível no serviço das gravações automáticas, mais conhecido como DVR.

Tendo em conta o intervalo de tempo disponível para a visualização de cada um dos diferentes tipos de conteúdos, quando se notifica o utilizador sobre um conteúdo ou se apresenta a página "Meus Conteúdos", é necessário confirmar previamente se este(s) conteúdo(s) ainda existe(m). Para este efeito, é realizado um pedido ao DVR, que contém todos os conteúdos que saíram do EPG durante sete dias. Passando desse limite, os conteúdos deixam de estar disponíveis, tornando a sua visualização inexecutável.

Para saber se o utilizador tem filmes alugados, é necessário recorrer ao serviço de alugueres do Videoclube. Sempre que existirem conteúdos deste tipo, como são do tipo VoD (*Video on Demand*), não é possível fazer uma consulta no EPG ou no DVR. Para se ter acesso à informação alternativa, é necessário recorrer ao motor de pesquisa do MEO utilizando o título do filme como variável de pesquisa. A razão deve-se ao facto de que os conteúdos VoD não estão disponíveis na programação linear.

Tendo em consideração que a funcionalidade de pedir para ver um conteúdo mais tarde, com recurso a linguagem natural traz um nível de personalização para um serviço já existente no MEO - Gravações Automáticas – não se pode descurar os conteúdos guardados através da mesma. Esta funcionalidade não necessita de interação por parte do utilizador, servindo o propósito entendido, na medida que guarda automaticamente um conteúdo cuja visualização está incompleta,

complementando o cenário de uso, quando o utilizador desliga a televisão ou adormece, ao obter o último conteúdo visto pelo utilizador.

Assim que as fontes de dados que alimentam a aplicação foram definidas e uma vez que nem todas as propriedades que são necessárias para apresentar conteúdos na televisão partilhavam o mesmo nome, ou apresentavam estruturas de dados semelhantes, definiram-se quatro modelos de dados, uma para cada tipo de estrutura de dados diferente: para pesquisas no EPG e no DVR, para pesquisas no serviço de alugueres do Videoclube e para as *Bookmarks* MEO, para pesquisas no *search engine* e por fim, para a pesquisa na base de dados Firebase Database.

5.1.2. Firebase Database

Uma vez que o serviço do MEO não tem um *backend* preparado para guardar conteúdos que o utilizador segue, marcou para ver mais tarde ou agendou, bem como, os dados necessários para o *Bot Framework* comunicar proactivamente com o utilizador, fez-se uma análise sobre o tipo de base de dados a que melhor acomodaria este projeto.

A base de dados Firebase foi definida para guardar os conteúdos em cima mencionados, por não necessitar de se desenvolver uma estrutura de base de dados de raiz, como nas bases de dados em SQL[9], por estar hospedada na nuvem e pelos dados serem armazenados como JSON[10] e sincronizados em tempo real para todos os clientes que utilizem esta base de dados. Como se pode observar na (Figura 15), os conteúdos que são guardados estão associados a um *Globally Unique Identifier* (GUID) da STB MEO utilizada e tem um identificador único gerado através do firebase com base na data de publicação, de forma a garantir que os dados não são substituídos. Para efeitos de prototipagem, o GUID da STB MEO é conhecido e, como tal, não se está a assegurar a devida privacidade e anonimato dos utilizadores.

[9] *Structured Query Language* é utilizada para aceder e manipular de bases de dados.
[10] Padrão aberto de formato de arquivo que usa texto legível para transmitir de forma assíncrona, objetos de dados em pares de atributo-valor e/ou dados em matrizes.

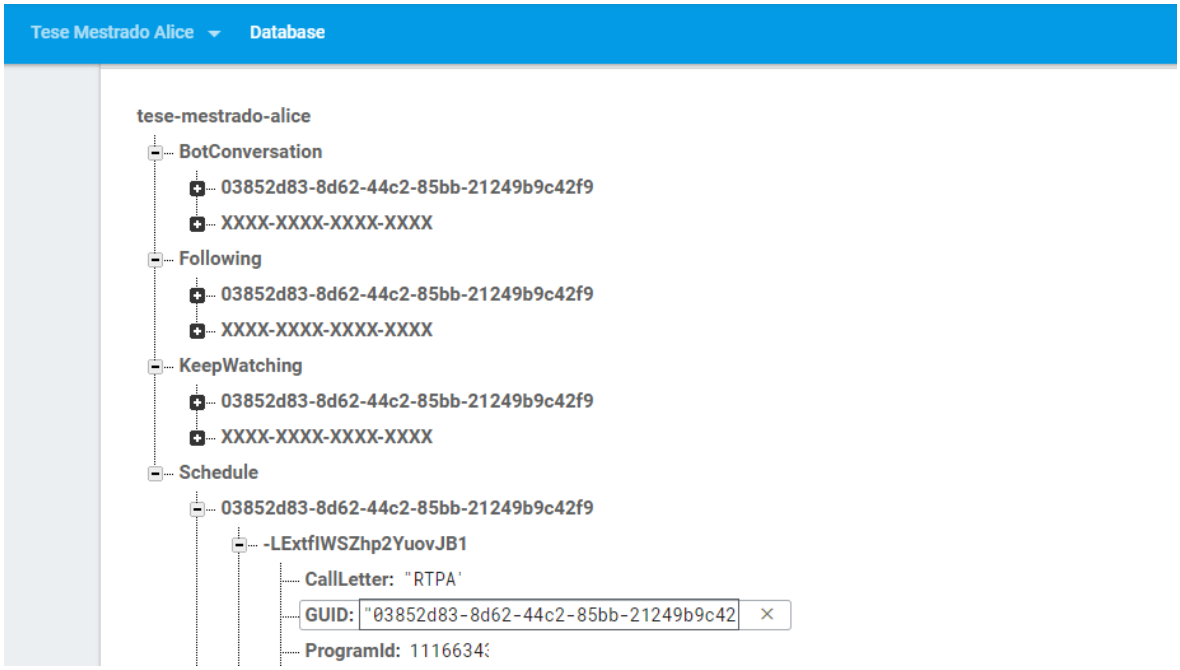


Figura 15 - Estrutura da Base de dados no Firebase

Dado que a proteção de dados reflecte um tema de preocupação ao nível global e tal não poderia ser descurado, uma outra das razões que levou à escolha desta base de dados foi, o acesso impedido mediante a inserção de credenciais incorretas através de uma integração do Firebase Authorization. Este serviço de autenticação é um serviço nativo do Firebase que permite a utilização de um GUID como forma de autenticação (em produção, o GUID armazenado nunca poderá ser o que diz respeito à *Box*). Ainda outra razão é a garantia que os dados são sempre armazenados com sucesso, uma vez que o SDK[11] do Firebase Database mantém os pedidos em disco, no caso de não haver conectividade, graças às bibliotecas de cliente poderem executar consultas do tipo ad-hoc.

Uma das desvantagens deste serviço é que, à medida que os dados aumentam, o desempenho da consulta começa a ser degradado. No entanto, se o Firebase souber quais são as chaves que se pretendem consultar, pode indexá-las no servidor, melhorando assim o desempenho das consultas.

Para isso é necessário definir um conjunto de regras (Figura 16), que para além de dar permissões de leitura e escrita (que necessitam de autenticação) aos diferentes nós criados (podemos observar na figura alguns desses nós, o "KeepWatching" e "Following"), é possível validar o tipo de dados e também, introduzir as variáveis chave que se pretende indexar. Deste modo, fez-se uma análise das consultas à base de dados e definiram-se as respetivas regras, informando o Firebase das chaves a indexar.

[11] *Kit* de desenvolvimento de *Software*

```
{
  "rules": {
    ".read": "auth != null",
    ".write": "auth != null",
      "KeepWatching": {
      "$uid": {
      ".indexOn": "EpgId"
      }
      },
        "Following": {
      "$uid": {
      ".indexOn": "ProgramId"
      }
```

Figura 16 - Exemplo de regras definidas para a base de dados Firebase

Para tirar o máximo de desempenho através da integração desta base de dados, neste projeto, fez-se uma análise do tipo de dados que seria necessário, para apresentar um resultado nas interfaces. Através de sucessivos testes, chegou-se à conclusão que se poupava mais tempo e recursos ao guardar toda a informação necessária no momento da interação (para seguir uma série, agendar ou ver um conteúdo mais tarde), criando uma lista de conteúdos, do que guardar apenas o identificador de cada conteúdo e fazer os pedidos de informação à posteriori, quando for para apresentar os resultados. Desta forma, todos os dados que são guardados nesta base de dados já foram processados pela API Proactive e correspondem à informação necessária para serem apresentados na televisão. Ao se proceder deste modo, são poupados preciosos segundos para dar feedback ao utilizador após a sua interação por voz, uma vez que o processamento da voz já leva um tempo considerável.

5.1.3. Componentes da API

Com a funcionalidade de Catch Up TV[12], os utilizadores já conseguem recuperar episódios que não conseguiram ver na televisão linear. No entanto, com este desenvolvimento, acresceram-se as validações de conteúdos, que são importantes para o sistema determinar se um conteúdo expirou, ou se está disponível para ser apresentado. Com base nessa necessidade e na eventualidade dos conteúdos que foram guardados na base de Dados Firebase terem expirado, todos os componentes que recolhem aquilo que o utilizador marcou para ver mais tarde, agendou ou o sistema notifica proactivamente, contêm uma validação junto das respetivas fontes de informação.

[12] Serviço de IPTV que permite ver conteúdos por um período limitado que passaram na televisão. Suporta também funcionalidades de manipulação da barra de progresso do conteúdo (pausa, rebobinar, entre outros).

Já os conteúdos que o utilizador segue sofrem uma abordagem diferente. Quando o utilizador pede para seguir um conteúdo, faz-se uma consulta às fontes de informação, semelhante a todos os tipos de conteúdo, para recolher o título, o identificador do programa, o canal de emissão, entre outros. Para garantir a obtenção da imagem de capa, é feita uma concatenação entre o link para aceder à imagem, o título e canal de emissão. Já quando se trata de séries, a principal diferença é que a série não expira, mas sim os seus episódios, sendo necessário guardar o identificador da série, para que mais tarde se possa validar a existência de episódios disponíveis no EPG ou nas GA (Gravações Automáticas). Mediante a existência de um conteúdo de uma série, é incluída a informação série na resposta do pedido dos que o utilizador segue.

Desta forma, garante-se que todos os conteúdos apresentados ao utilizador estão disponíveis e previne-se também que haja conteúdos repetidos. Caso se pretenda guardar um conteúdo com um identificador que se encontre na base de dados, observam-se as datas previamente definidas para aferir se expiraram, atualizando a informação do conteúdo com o identificador em causa.

Para haver comunicação com a base de dados Firebase, fez-se um estudo de como integrar uma livraria de Firebase que funcione com a linguagem de programação da Microsoft .NET, visto que o Firebase pertence à Google e não possui uma forma de o integrar diretamente com esta linguagem de programação. Fruto desse estudo foram descobertas três bibliotecas abertas que integram o Firebase com .NET, a livraria "FireSharp", a "FirebaseSharp" e a "firebase-database-dotnet".

Ao se integrar a livraria "FireSharp" na API Proactive, encontrou-se um problema com os resultados que se obtinham dos pedidos à base de dados, uma vez que o resultado obtido desta livraria é a concatenação do caminho onde foi recolhida a informação com a resposta, ambas encapsuladas numa *string*. A livraria "FirebaseSharp" não permite o uso de filtros para obter resultados, impedindo assim o acesso à informação específica de um conteúdo. A livraria que satisfazia todos os requisitos necessários para este projeto é a "firebase-database-dotnet", eliminando todos os inconvenientes das livrarias anteriores. Um dos aspetos positivos, é que a sua integração é feita de forma simples e semelhante à de outras linguagens de programação suportadas pelo Firebase, que já tinham sido aprendidas.

Um dos objetivos que se pretendia concretizar com este protótipo era a possibilidade de o utilizador ser notificado quando existe um conteúdo do seu interesse. Após se analisarem as abordagens possíveis para se apresentar conteúdos ao utilizador, sob a forma de uma notificação, implementaram-se dois tipos de notificação: com conteúdos agendados a pedido do utilizador e gerados de forma automática pelo sistema.

As notificações que são geradas a pedido do utilizador tratam-se de conteúdos agendados para uma data específica, que foi validada conforme o tempo de disponibilidade do respetivo conteúdo nas GA's. Uma vez que o Mediaroom e os serviços MEO não têm sistema de notificações desenvolvido e tendo em ponderação que gerir as notificações do lado da STB, ou do *Hands Free Device,* pode levar à perda das mesmas (o utilizador pode desligar os dispositivos impossibilitando o controlo de notificações), criou-se uma cache do lado da API por agendamento, cujo tempo de expiração dessa cache coincide com a data que o utilizador quer ser notificado. Quando chegar esse

momento, o sistema executa uma página no Mediaroom para abrir a aplicação de notificações com o respetivo conteúdo agendado.

Se o utilizador segue alguma(s) série(s), quando a STB MEO é ligada, é feita uma verificação se existem conteúdos relativos que vão expirar, através da Companion API (como explicado no subcapítulo 5.5). Dado que o ambiente MEO contém conteúdos que permanecem temporariamente (com uma disponibilidade de sete dias) nas GA, é necessário que haja um componente que priorize os conteúdos com base na sua disponibilidade, para notificar o utilizador. Esta prioridade sob a qual os conteúdos são apresentados, leva a seguinte ordem em consideração:

I. o cliente alugou algum conteúdo no Videoclube e não o finalizou;
II. o utilizador desligou a televisão sem acabar de ver um conteúdo;
III. existe um conteúdo que o utilizador segue e que vai expirar;
IV. o conteúdo que o utilizador marcou para ver mais tarde vai expirar;

Uma vez que os conteúdos podem ser apresentados várias vezes ao longo do dia – e o conteúdo aparece no EPG consoante o número de vezes que vai ser reproduzido num dia, - o sistema guarda a data do último programa que for apresentado mais tarde, para priorizar os conteúdos conforme aqueles que expiram primeiro. Por outras palavras, se um filme no canal Fox der na parte da manhã e repetir à tarde, o sistema vai considerar a data da tarde como a data inicial, para obter o maior intervalo de tempo do conteúdo que vai estar disponível por sete dias nas GA.

Para apresentar séries que o utilizador segue e conteúdos que marcou para ver mais tarde na área "Meus Conteúdos", foram desenvolvidos vários componentes que alimentam a estrutura de dados pretendida. Ao separar estes componentes por funções mais simples e orientados para um objetivo específico, podem ser reaproveitados noutras partes da aplicação. Destacam-se os métodos de recolha de informação que são responsáveis por gerir a disponibilidade da diversidade de conteúdos que a aplicação consome. Desta forma, os únicos métodos que não são recicláveis, são aqueles responsáveis por criar a estrutura de informação que cada página espera receber.

5.2. NLU Microsoft LUIS

Para se obter a intenção necessária para executar um comando na aplicação Mediaroom, de todas as interações por voz, é importante definir os *intents* ou intenções e as respetivas frases (*utterances*) nos diálogos que o utilizador pode dizer. Desta forma, por cada *intent* definido, o LUIS tem de ser treinado com as varias frases diferentes, que convergem para a mesma intenção, como se pode observar na Figura 17. Apesar do LUIS conseguir compreender nuances entre as *utterances* treinadas e semelhantes ditas pelo utilizador, conseguindo atingir os *intents* previstos, diferenças mais evidentes necessitam que o LUIS seja treinado para as conseguir compreender como parte daquele *intent*. Sempre que o LUIS não entende uma frase, ele guarda-a num *BackOffice* para que seja possível ser adicionada ao *intent* a que diz respeito.

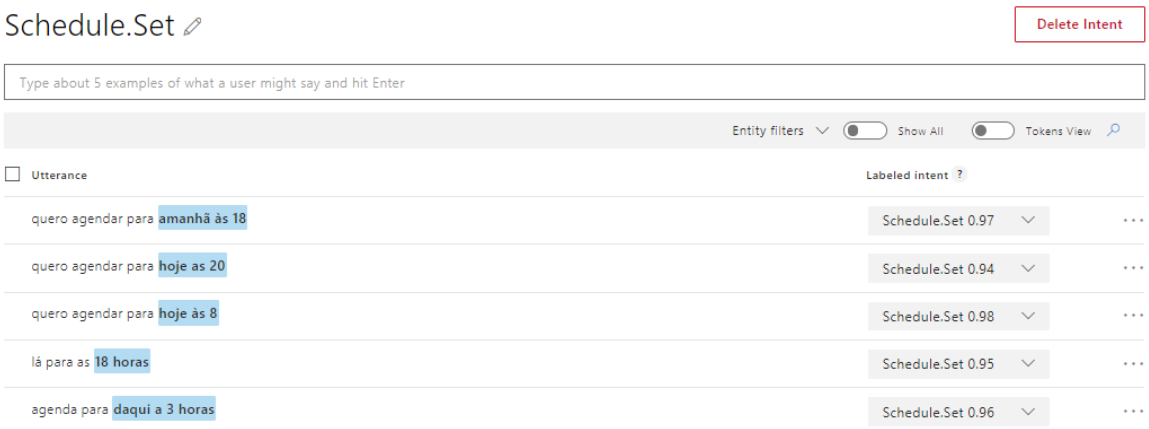


Figura 17 - Utterances do intent "Schedule.Set"

Os *intents,* por si só, nem sempre conseguem definir o que é que o utilizador pretende. Por outras palavras, o utilizador até pode atingir a intenção de agendar um conteúdo. No entanto, o sistema precisa de saber qual é a data que ele quis agendar. Para dar resposta a esta necessidade, assente em várias intenções, o LUIS permite a criação de *Entities* ou entidades, que podem ser definidas pelo programador, ou pré-definidas pelo LUIS, já com algum tratamento de informação, como o caso de números e datas. Assim, as *entities* podem ser palavras ou conjuntos de palavras, que definem variáveis importantes para a execução de uma tarefa e que deriva da intenção do utilizador. A parte da *utterance* que diz respeito a uma *entity* é o retângulo em azul, presente nas várias frases da Figura 17.

Uma vez que estre protótipo está assente numa base já contruída pela Altice Labs, fez-se um estudo sobre as melhores práticas no uso do NLU LUIS, quando existem *intents* de temáticas diferentes. Recorrendo à documentação do Microsoft LUIS para esta análise concluiu-se que, como o LUIS tem um limite de mil *intents* por aplicação e como os *intents* que a Altice Labs desenvolveu tinham objetivos diferentes do que é pretendido para este protótipo, adotou-se uma estratégia comummente recomendada para este tipo de situações, que é a divisão de apps para conter *intents* específicos. Deste modo, a aplicação [13]que a Altice Labs construiu inicialmente foi dividida em duas aplicações, a "AliceVodsContext" que contém *intents* específicos para a pesquisa de conteúdos VoD e a "Alice" que se trata de uma aplicação base, contendo todos os *intents* principais de cada aplicação LUIS, bem como, comandos mais simples que se traduzem na execução imediata de uma ação na aplicação Mediaroom.

[13] A aplicação LUIS tem o nome de Alice por ser a primeiro componente desenvolvido. As restantes aplicações adotaram o nome de Sofia.

Para dar resposta às necessidades deste protótipo e seguindo a mesma conformidade, criou-se a aplicação "ProactiveContext" no LUIS com catorze *intents*. O desenvolvimento de uma aplicação LUIS é um processo iterativo e decompõem-se em 5 etapas conhecidas como aprendizagem ativa[14]:

I. Definir o *schema* de intents*;*
II. Treinar o *schema* com *intents* e *entities*;
III. Adicionar *phrase lists* ou *patterns;*
IV. Testar os *intents*;
V. Publicar a App LUIS;

Na primeira etapa, mapeiam-se as intenções possíveis. Após definidas as intenções, treina-se o LUIS com um conjunto de frases (o recomendado pela Microsoft são dez frases) e adicionam-se as entidades que se pretendem extrair de cada *utterance.* Para melhorar a performance do LUIS, é encorajada a utilização de *patterns*, que por outras palavras, são *templates* de frases que poderão ser ditas pelo utilizador e que dizem respeito a um dado *intent.* As *phrase lists* são grupos de palavras ou frases que pertencem à mesma classe e que devem ser tratadas de forma semelhante. Quando a etapa II. e III. estão completas, procede-se a uma bateria de testes com várias frases para verificar se o *intent* que o LUIS devolve está de acordo com o que foi planificado. Por fim, para utilizar as alterações deve-se proceder à publicação da aplicação nos servidores da Microsoft Azure[15]. É de salientar que o processo que se segue desde a II. etapa até à etapa de publicação é um ciclo que se repete múltiplas vezes ao longo do desenvolvimento, mesmo para *intents* que se pensavam fechados. Sempre que se faz uma alteração, é necessário voltar a treinar o LUIS e, se não ocorrerem conflitos, faz-se uma nova publicação.

Após mapear e treinar o LUIS com todos os *intents* e *entities* que constam na aplicação, procedeu-se a uma análise dos *intents* que também devem ser mapeados para a app "Alice" e repetindo o mesmo ciclo, foram adicionados apenas aqueles que iniciam um novo diálogo.

No inicio do desenvolvimento, como as aplicações não estavam segmentadas para aumentar a sua eficiência, quando se faziam os testes para treinar o LUIS, surgiam por vezes alguns conflitos entre outros *intents* que já tinham sido criados, por conterem palavras iguais, mesmo que para o ser humano, se traduzissem em intenções claramente diferentes. Ao adotar a estratégia divisão de aplicações LUIS, verificou-se um aumento significativo no valor obtido para cada *intent* na aplicação "Alice", através da ferramenta de testes do NLU e menor confusão entre *intents*, que outrora eram semelhantes na sua construção frásica. Esta solução também favorece o fluxo do serviço Azure

[14] https://docs.microsoft.com/en-us/azure/cognitive-services/luis/home
[15] Plataforma aberta com serviços de computação em cloud.

Bot[16] (conforme descrito no próximo subcapítulo), que é utilizado para fazer a gestão das conversas entre o utilizador e o sistema.

5.3. Azure Bot Service

Para que seja possível manter um diálogo com a televisão, o sistema precisa de algum mecanismo para poder guardar a informação que vai recolhendo do utilizador, saber o estado da conversa à medida que ela vai progredindo e ir respondendo em concordância com o que é dito pelo mesmo. Dada essa necessidade e tomando em consideração que o NLU utilizado no projeto é da Microsoft, o Bot que foi escolhido para este protótipo foi o Azure Bot, considerado um dos melhores, garante um bom suporte ao desenvolvimento e permite integração direta dos serviços cognitivos do LUIS, sendo disponibilizados, inclusive, *templates* para reduzir o tempo na integração de ambos.

Um serviço de Bot tem a função de correr tarefas simples e repetitivas de forma automática pela internet, a enormes velocidades. Um *Dialog* Bot como o Azure Bot, não é indiferente a este tipo de trabalho, sendo capaz de mediar o estado de múltiplas conversas entre os sistemas e seus utilizadores. Para que possa haver interação por linguagem natural, o sistema tem de ter a capacidade de entender a intenção do texto do que o utilizador disse. É nesta fase que, através da integração do NLU LUIS, o *Bot* consegue extrair a intenção do texto para devolver uma resposta de acordo com o *intent* que obteve a maior pontuação no LUIS.

Como demonstra a Figura 18, o sistema está montado para que sempre que o utilizador inicia uma nova conversa com o sistema, o resultado dessa interação é enviado para uma página "Root"[17] de diálogo no Bot, onde se verifica a existência do *intent* na aplicação "Alice" do NLU. Ao nível da estrutura, a "Alice" é utilizada para verificar os *intents* iniciais, que caso coincidam com os da app "AliceVodsContext" ou "ProactiveContext", o Bot instancia novos diálogos que passam a fazer a obtenção dos *intents* na aplicação LUIS mais apropriada.

[16] Serviço do Microsoft Azure para a criação de uma interface conversacional através de um robot com diálogos pré-programados.
[17] Também denominada de página principal.

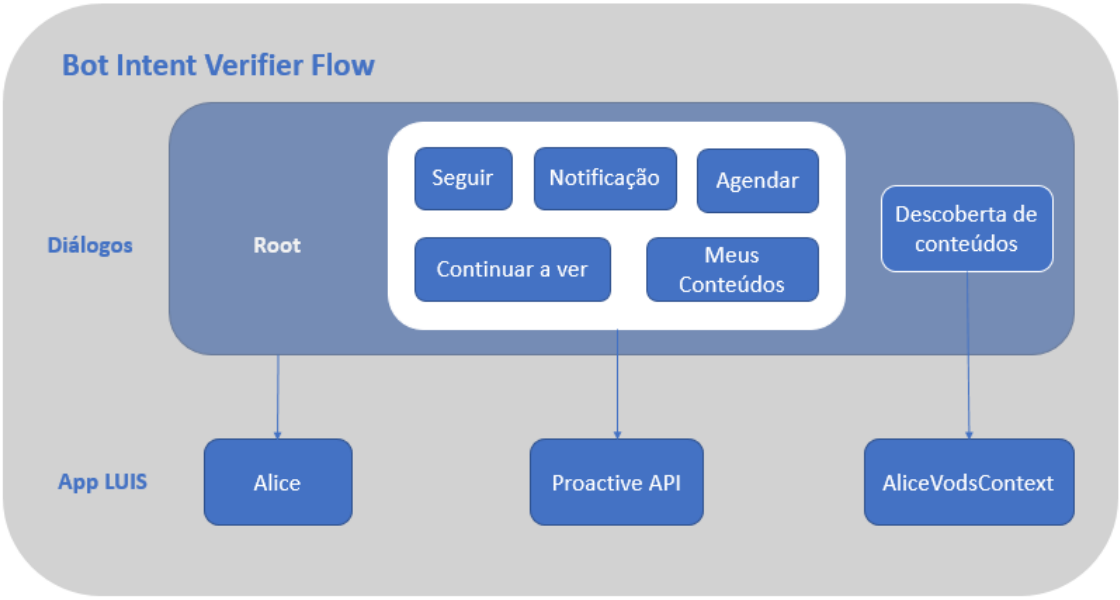


Figura 18 - Estrutura de verificação dos intents para os diálogos

Sempre que for iniciada uma conversa no *Bot*, existe um compasso de espera para o utilizador dar uma resposta. Esta interação termina quando o diálogo chega ao fim, ou é interrompida pelo utilizador. Existem algumas exceções para os *intents* mais simples, como o caso de mudar de canal na televisão, que dão uma resposta imediata e, por conseguinte, o diálogo é fechado.

Sempre que um diálogo chegar ao fim, é executado um conjunto de métodos para produzir uma resposta com dados, nomeadamente o nome do *intent*, a frase dita pelo utilizador, o texto de resposta gerado pelo *Bot,* o texto que o componente de text to *speech* Polly vai falar e *entities* que resultaram dessa interação. Por vezes, as *entities* pré-contruídas pelo LUIS, como o caso da data, nem sempre têm o mesmo nome da variável que contém a data. Isto deve-se ao facto de que o NLU consegue interpretar se o utilizador se está a referir-se a uma data, a um intervalo ou a um número específico. Deste modo, é necessário refinar os dados para serem utilizados à posteriori, no contexto da mensagem e em linguagem natural. Por outras palavras, utilizando a intenção de agendar um conteúdo, se o utilizador quiser agendar para trinta minutos após fazer o pedido, o sistema vai produzir uma mensagem do tipo “Vou agendar para daqui a 30 minutos, ok?”. Caso se trate de uma hora específica no próprio dia ou no dia seguinte, o sistema interpreta a data como “hoje” ou “amanhã”, seguido apenas da hora à qual vai lembrar o utilizador, caso contrário, segue o padrão normal com a data por extenso. Estas são algumas das nuances que são geradas para dar ao utilizador a sensação de que está a comunicar com outro ser humano e para considerar o sistema inteligente.

O resultado final contruído pelo *Bot* é enviado para a aplicação Node Sofia, que reencaminha a resposta para o Companion e como se explica no subcapítulo seguinte.

5.4. Aplicação em Node Sofia

A aplicação em Node Javascript Sofia pertence à base do sistema previamente desenvolvido pela Altice Labs. Visto que esta aplicação tem como objetivo a gestão do fluxo da interação por voz, não houve necessidade de acrescentar algum contributo, uma vez que já serve o seu propósito neste protótipo. Não obstante, o fluxo da interação por voz que é feito através desta aplicação pode ser consultado no subcapítulo 4 e é considerada um dos pilares deste projeto, por conectar a maioria das aplicações onde foi dedicado um maior investimento no desenvolvimento de funcionalidades.

Para fazer testes durante a implementação dos vários componentes, foi necessário modificar os ficheiros de configuração desta aplicação. Estes contêm todas as variáveis como o IP da STB, GUID da STB, chave de segurança do *Bot*, o *url* da API Companion, entre outras. A razão, pela qual estas variáveis estão contidas neste ficheiro, é porque como é necessário compilar a aplicação funcionar e, como se faz uma *build* do projeto inteiro, o único ficheiro que se pode modificar é este, todas as variáveis que poderão mudar ao longo do tempo são alojadas neste ficheiro. Para os se proceder aos testes de funcionalidades e para o *Deploy* do projeto, quase todos os valores se mantiveram imutáveis, com exceção do IP da STB e o segredo do Bot. Isto, devido a mudanças de IP na rede e ao desdobramento do *Bot* para uma versão de desenvolvimento e outra de produção, com o objetivo de manter sempre uma versão estabilizada para fazer Demos do protótipo, sem interrupções causadas por um *sprint* de implementação.

5.5. Companion API

A Aplicação Companion surge como um mediador entre a informação obtida através do *Dialog Bot* e aquela que a aplicação Sofia Mediaroom está à espera para apresentar o *feedback* da interação por voz e para concretizar as ações pretendidas. No entanto, faz-se todo um processo de gestão e processamento de *intents* nesta aplicação, que seguem várias etapas até se materializarem em eventos a ser executados na aplicação Mediaroom.

Depois de obtidos os resultados pelo *Bot,* o Companion recebe a resposta através do Node Sofia e verifica se existe uma correspondência na sua *stack* de *intents* mapeados. Caso se confirme uma conexão, a API reencaminha a informação para o respetivo método, responsável por construir um evento para executar um comando no Mediaroom. De um modo geral, o evento contém o nome do *intent*, o GUID da STB, a mensagem do utilizador e do *Bot,* o endereço IP do Raspberry e a mensagem que vai ser reproduzida pelo componente de *text to Speech* Polly.

Tendo em conta que se pretende melhorar a experiência de televisão, através da criação de funcionalidades que sejam proactivas e facilitem o acesso a conteúdos do interesse do utilizador, o Companion envia uma notificação para o televisor com conteúdos que acha que o utilizador pode querer retomar ou que estejam para expirar.

As notificações, geradas através da API Companion, surgem com recurso à mesma página Mediaroom que o utilizador recebe quando agenda um conteúdo. A principal diferença é que em vez da API criar uma *cache*, a notificação é despoletada através um método no arranque da API Companion, que corre num *Loop*. Quando se verifica que a STB foi ligada e saiu do modo de

descanso, é criada uma tarefa que faz um pedido à Proactive API, descrito no subcapítulo 5.1.3, para aferir se existe um conteúdo que possa interessar ao utilizador.

5.6. Aplicação Sofia Mediaroom Presentation Framework

O Mediaroom é um *midleware* que permite a comunicação nos dois sentidos (permite interação do utilizador com o sistema) e oferece um conjunto de serviços por subscrição de IPTV como conteúdos protegidos por DRM, *live*, gravações, *video on demand, multiscreen e applications.* Este *middleware* é o serviço que o MEO tem nas suas STB. Para desenvolver aplicações rápidas e elegantes, recorreu-se ao SDK Mediaroom Presentation Framework (PF). Este SDK permite que as aplicações nele desenvolvidas sejam aprovisionadas através de serviços Web, com uma lógica de código semelhante à construção de uma página web. No entanto, a capacidade lógica é mais limitada e são precisos conhecimentos sólidos de programação para se produzirem aplicações com esta tecnologia. Não obstante, estes aspetos negativos não causaram impacto para o desenvolvimento e integração com os serviços de interação por voz, tendo sido apenas fundamental acautelar alguns pormenores, que são analisados em detalhe ao longo do capítulo.

5.6.1. Pagina Mediaroom Inicial da Aplicação Sofia

A interação por linguagem natural é um avanço significativo ao acesso rápido a informação, face à interação com o telecomando e mais permissiva que comandos por voz. No entanto, esta interação por si só pode gerar frustração no utilizador, porque, como os serviços de voz funcionam através da *cloud* e dependendo da largura de banda do utilizador, o processamento de voz pode levar um tempo considerável para se obter resposta. Face a este tipo de problemas de interação por voz que foram enunciados no estado da arte, foi desenvolvida uma página Mediaroom que surge sempre que é ativada a *wake up word*, com uma mensagem de feedback numa caixa de texto (Figura 19), que é modificada à medida que o diálogo evolui. Para além disso, a interface mostra alguns *hints* visuais no ícone do microfone para que o utilizador possa saber quando é que a Sofia está a ouvir.

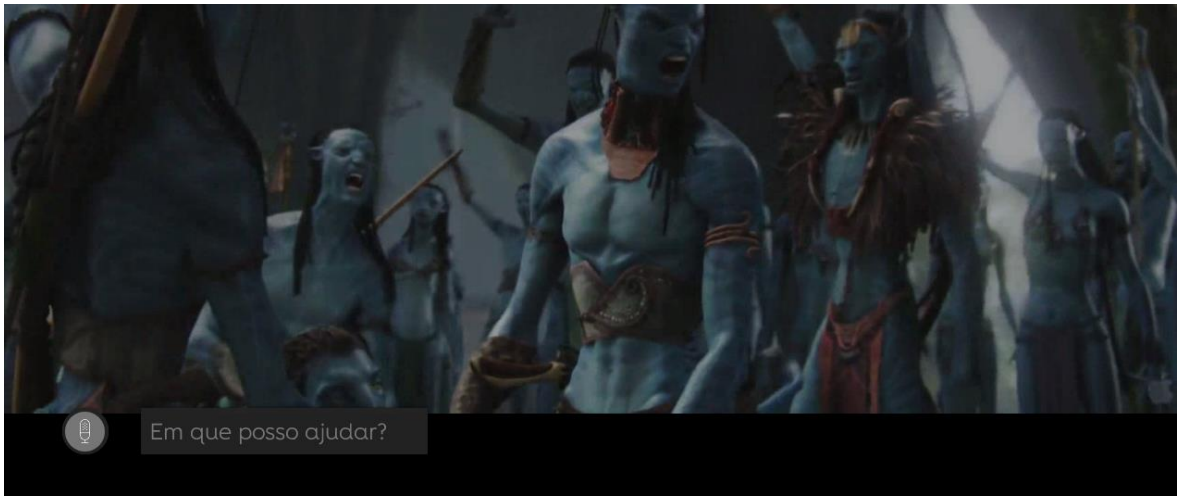


Figura 19 - Pagina Mediaroom invocada pela *wake word* "Sofia"

A interface, que marca o início da interação, tem um grande valor no protótipo porque é onde se apresenta o feedback de todas as interações, seja durante um diálogo ou seja para execução de tarefas imediatas. No entanto, a contribuição efetuada nesta página, foi ao nível da integração das funcionalidades de diálogo e acesso à pagina de "Meus Conteúdos". Como podemos observar na Figura 20, as diferenças começam a ser evidentes, quando o utilizador entra em diálogo com a Sofia, pelo contraste que existe entre falar e despoletar um comando, com o que existia previamente *versus* entrar num diálogo com o sistema e, mediante a intenção, receber uma mensagem personalizada do sistema, para o utilizador.

Figura 20 - Diferentes estágios da *label* que apresenta o *Feedback* da Interação por voz

Ultrapassaram-se vários desafios relativos a como é que a interface se deveria comportar para se adaptar ao diálogo, levando dois fatores em consideração: não consumir uma grande área da emissão que corre em segundo plano e manter uma fácil distinção entre o que foi dito pelo utilizador e pela Sofia. Para tal, fizeram-se várias reuniões com um grupo de *experts* em *interfaces* gráficas para televisão na Altice Labs, tendo sido discutida, para esta página em específico, o tamanho da caixa de texto da mensagem, a utilização de cores ou duas caixas de texto diferentes para distinguir as mensagens, entre outras. Uma vez que poderiam existir complicações no futuro ao modificar a *interface* para duas caixas de texto que simulavam um *look* ao estilo corrido de perguntas e respostas, optou-se por utilizar a distinção das mensagens através da cor. Esta solução aparentava ser uma implementação fácil e rápida, uma vez que o Companion envia um evento com as mensagens do utilizador e do Bot separadas. No entanto, existe uma limitação ao nível do Mediaroom que, apesar das *labels* terem uma cor base e uma cor *highlight*, é necessário colocar um código entre a mensagem que fica com a cor *highlight*, sendo apenas interpretado na página ASPX (Active Server Page Extended). Visto que todos os eventos funcionam com recurso a uma camada de Javascript, o código inserido no texto na *label* da mensagem é lido como parte da *string*. Para resolver este problema, implementou-se uma *label* por cima da que já existia, com o texto de cor azul, sendo determinada programaticamente, a parte do texto correspondente à parte do Utilizador, como demonstrado na Figura 20. Já a Figura 21 mostra que, se o tamanho da

concatenação das duas mensagens for maior do que o espaço da *label* principal, o texto é cortado no princípio, destacando a mensagem do fim, que diz respeito ao Bot. Embora não seja uma solução muito elegante, acabou por ser a melhor alternativa a uma implementação mais complexa, visto que a página está assente numa *layer* da aplicação Mediaroom que se sobrepõe a tudo o que estiver a ser mostrado no ecrã e serve para mostrar páginas que ocupam pouco espaço em memória.

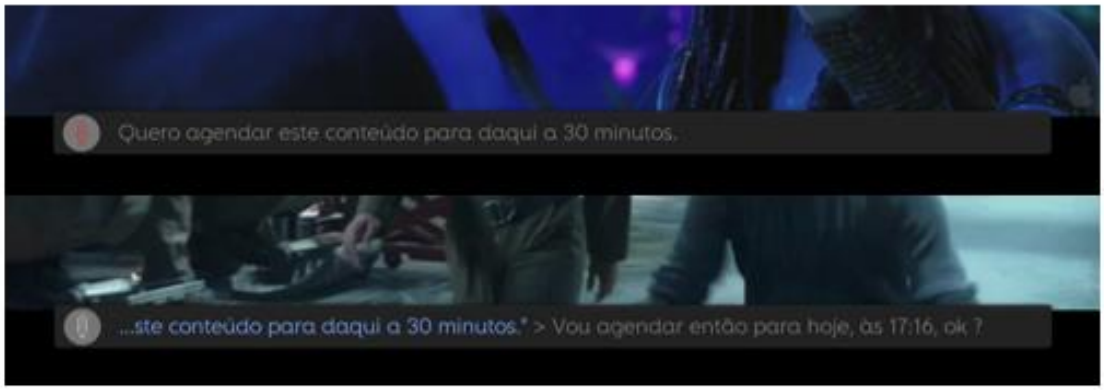


Figura 21 - Exemplo de um *overflow* de texto

5.6.2. Pagina Mediaroom "Meus Conteúdos"

A página de "Meus Conteúdos" tem o propósito de apresentar as séries que o utilizador segue e os conteúdos que marcou para ver mais tarde. A *interface* apresenta uma área retangular com um *look and feel* semelhante ao MEO com duas listas horizontais, em que a do topo diz respeito às séries que o utilizador segue, com o título "as minhas séries", e a última lista têm o título "Continuar a Ver" que, tal como indicia, apresenta os conteúdos que o utilizador marcou para ver mais tarde. Quando se permuta entre listas, a UI adapta-se, animando a posição e o tamanho do título da lista selecionada, de forma a manter a coesão entre as margens do título e da Imagem que passa a ser destacada.

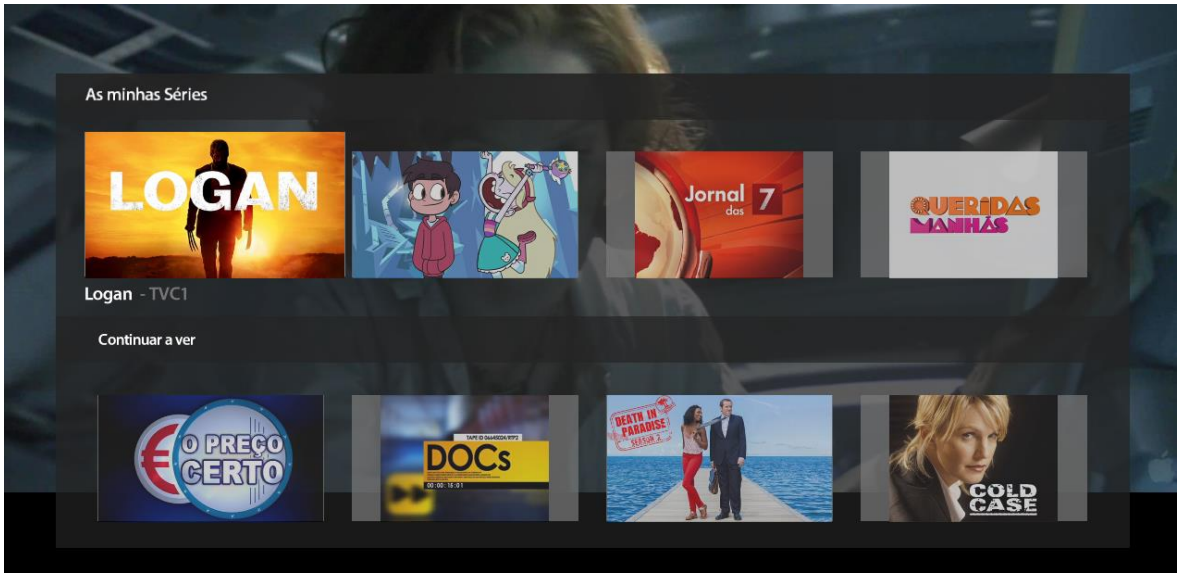


Figura 22 - Interface "Meus Conteúdos"

As séries que são apresentadas na linha "As minhas Séries" são exclusivamente conteúdos que estavam a dar *Live* ou nas Gravações automáticas e que o utilizador pediu para seguir. Significa que, a partir desse momento, as capas dos conteúdos que aparecerem nesta área, contêm episódios que o utilizador pode ver. Quer isto dizer que se a API Proactive verificar que não existe um conteúdo disponível naquele momento para ser visualizado, a capa não é apresentada nesta *interface*. Quando o utilizador pressionar o botão de "OK" do telecomando numa capa desta lista, é reencaminhado para uma página de detalhe do conteúdo no ambiente MEO, onde pode escolher os episódios que quer ver, ou saber mais informação sobre a série que segue.

Os conteúdos que são apresentados na lista "continuar a ver" estão ordenados por conteúdos que o utilizador marcou para ver mais tarde e o continuar a ver automático do MEO. Esta ordenação pressupõe que ambas as fontes de informação apresentam o mesmo tipo de conteúdos, assumindo que a prioridade seja com base na interação do utilizador. Quando o utilizador pressiona o botão "OK", o conteúdo é imediatamente reproduzido para momento em que tinha ficado anteriormente.

5.6.1. Página Mediaroom de notificações

Para além da página "Meus Conteúdos", foi necessário desenvolver uma interface de Notificações que albergasse conteúdos que o utilizador agendou e, também, aqueles que o sistema apresenta de forma proactiva. Sempre que for despoletado um evento no Mediaroom que invoque a página mencionada, aparece no ecrã um retângulo branco com pouca transparência que contém os *placeholders* para a informação do que se pretende notificar. Este retângulo surge através de uma animação horizontal que começa fora do ecrã, com sentido da direita para a esquerda. Como se pode observar na Figura 23, a notificação apresenta o título, o nome do canal e uma imagem do conteúdo. Enquanto a interface é contruída para ser mostrada na televisão, a informação a apresentar é carregada de forma assíncrona, com o auxílio do método *datasource* do Mediaroom PF, que está vinculado à estrutura que contém os *placeholders* para os dados do conteúdo,

atualizando-os dinamicamente à medida que são carregados.

Figura 23 - Notificação de um conteúdo agendado pelo utilizador

A Figura 23 mostra a notificação que é apresentada ao utilizador, quando se trata de um conteúdo agendado pelo mesmo. Por outras palavras, este cenário só vai acontecer se o utilizador estiver a ver televisão no momento em que o temporizador para mostrar a notificação expirar. No entanto, o conteúdo ficará disponível na área dos "Meus Conteúdos" até expirar o tempo nas gravações automáticas. Para se chegar a este *layout*, fizeram-se várias versões, que foram evoluindo de acordo com a aplicação MEO e conforme se foram identificando problemas de usabilidade no layout. O primeiro *layout* que se fez, apresenta uma interface gráfica com um *layout* semelhante aos menus da aplicação MEO, com *background* retangular escuro e cor do texto branca, como se pode verificar na Figura 24. No entanto, rapidamente se encontraram problemas com esta solução, pela confusão entre o contraste do vídeo e do background, sobreposição de menus (A notificação corre na camada superior da aplicação MEO), imagem do conteúdo pequena, e no caso de haver títulos com palavras compridas, dificilmente se conseguia perceber o conteúdo que estava a ser apresentado.

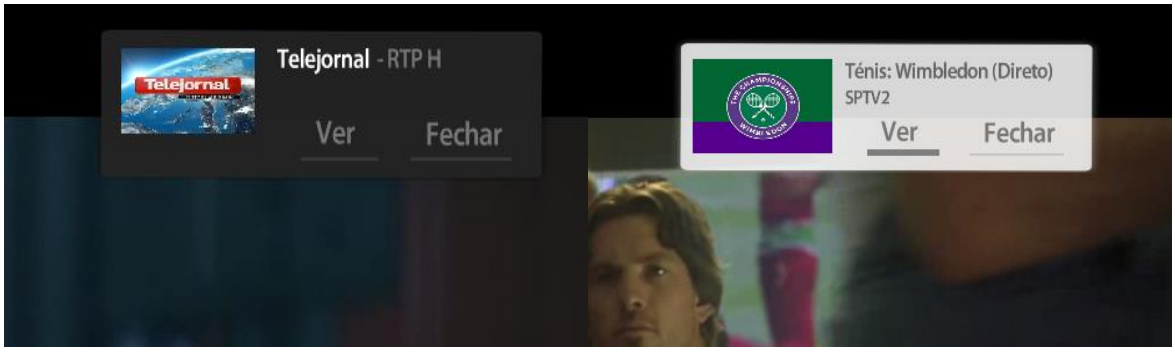


Figura 24 - Primeira versão do layout (esquerda) e versão final (direita)

Esta interface sofreu várias interações para corrigir estes problemas de usabilidade, dando origem a um *toaster* [18]de notificações branco, que se sobrepõe ao menu da aplicação MEO, e percetível com fundos mais claros. O título ganha maior destaque e a imagem foi aumentada, para que seja mais fácil de ser visualizada à distância, na televisão.

Com o desenvolvimento das notificações proactivas deste projeto, foi discutido na Altice Labs a possibilidade de o utilizador poder aceder aos "Meus Conteúdos" caso a primeira notificação que aparece, quando inicia a sua *box*, não seja o conteúdo que procura visualizar. Deste modo, fez-se uma restruturação do código das notificações, para adaptar o seu aspeto mediante uma *flag* de "Notification" ou "StartUpNotification", para que as notificações geradas dinamicamente pelo sistema apresentassem o layout proposto na Figura 25.

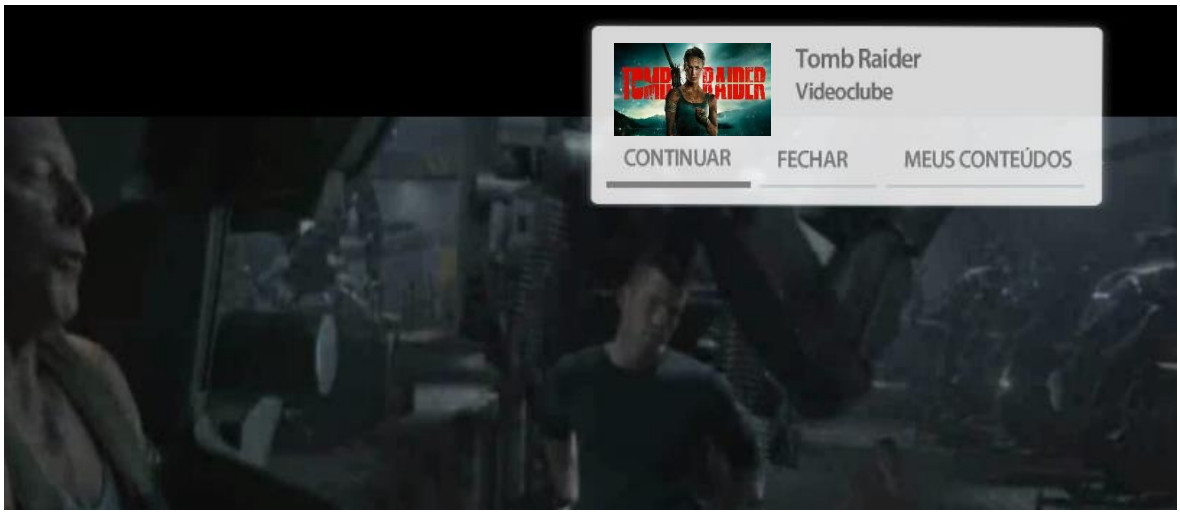


Figura 25 - Notificação gerada pelo sistema quando a STB é iniciada

Os conteúdos que surgem nas notificações quando a STB é iniciada, provêm do videoclube, que o utilizador não viu na sua plenitude na última sessão televisiva, conteúdos que vão expirar, e/ou por último, que marcou para ver mais tarde.

5.7. *Deploy* do protótipo

Para que sejam integradas todas as aplicações que estão presentes na Arquitetura do capítulo terceiro, houve uma necessidade de alojar algumas dessas aplicações nos servidores da Altice Labs, bem como nos servidores da Microsoft Azure. As aplicações que foram alojadas no servidor da Altice Labs foram: a Aplicação Sofia Mediaroom Presentation Framework e a Proactive API. As aplicações NLU Microsoft LUIS e Azure Bot Service, como foram desenvolvidas com os serviços do Azure, necessitam de ser alojadas nos servidores da Microsoft Azure. Todas estas aplicações

[18] Notificação descartável, cujo objetivo é informar o utilizador de algo, sem necessidade de interação do mesmo para a notificação desaparecer do ecrã.

mencionadas contêm um caminho fixo atribuído por cada servidor onde foram alojadas. Um dos benefícios de ter o NLU num servidor da Microsoft é que sempre que for necessário atualizá-lo, o LUIS atualiza de forma automática o servidor onde está alojado, com a informação mais recente.

No Raspberry Pi, foram alojadas a Aplicação em Node Sofia, a Companion API e uma aplicação simples que recebe o texto e converte num ficheiro de áudio, com recurso ao *text to speech* da Polly. Estas três aplicações são conectadas ao resto dos componentes deste projeto, através de um acesso *localhost*, com a respetiva porta, visto que ambos estão contidos no Raspberry PI, que por sua vez, encontra-se numa rede LAN, partilhada com a STB MEO.

IV. Recolha de dados e discussão dos resultados

A principal missão do trabalho aqui descrito é permitir aos utilizadores interagir através da voz com o sistema televisivo MEO, para seguirem, marcarem e/ou agendarem um conteúdo para ver mais tarde, com o intuito do sistema enviar notificações de forma proactiva, quando estes conteúdos estiverem a expirar. Para avaliar o protótipo proposto, bem como todas as implementações que foram efetuadas para cumprir esta missão, foi convidado um grupo de pessoas que trabalham na Altice Labs e um grupo de pessoas externas à empresa, para participar numa entrevista cujo objetivo foi comparar a facilidade de uso do telecomando com a interação por linguagem natural, quando se interage com o sistema MEO. Para além disso, pretende-se avaliar alguns aspetos nas *interfaces* que foram conceptualizadas para este protótipo, tendo sido validadas previamente por *experts* em design de *interfaces* para televisão da Altice Labs.

Uma das partes centrais deste projeto e objeto de estudo deste trabalho é desenvolver uma arquitetura que permita interagir por voz por linguagem natural que seja capaz de gerir um diálogo com o utilizador, para executar uma determinada ação no sistema MEO, relativa ao conteúdo que está a ser visualizado. Deste modo, deve-se ter atenção aos diferentes componentes que compõem a voz, bem como perceber como é que o resultado produzido por cada componente afeta a perceção do público-alvo sobre o sistema, sobretudo pelas nuances de interação verbal que existem na comunicação humana. Perceber a forma como os seres humanos interagem entre si por voz é muito importante para desenvolver um sistema robusto de interação por linguagem natural, havendo uma necessidade intrínseca de considerar quais são as palavras-chave que os utilizadores pensam sobre uma determinada ação que querem realizar e como articulam a mensagem, de modo que essa ação possa ser realizada pelo sistema.

1. Metodologia adotada para a recolha de dados

O momento de avaliação e validação do protótipo iniciou logo após o término da fase de prototipagem e desenvolvimento, para validar a viabilidade no contexto de utilização a que o protótipo se destina junto do público-alvo. O objetivo é perceber se a interação por voz através de linguagem natural é utilizada com mais facilidade do que com o telecomando e identificar princípios orientadores no desenvolvimento de uma aplicação de televisão, com interação por voz.

Para avaliar o protótipo proposto optou-se for realizar um estudo de caso explanatório. Para Yin (2001) o estudo de caso do tipo explanatório pretende identificar questões de pesquisa e possíveis abordagens para projetos futuros. O estudo de caso não segue uma abordagem rígida de investigação, permitindo a combinação dos três métodos qualitativos de recolha de informação mencionados no capítulo da metodologia, como os questionários, o guião de entrevista e a entrevista semiestruturada. Cada método qualitativo teve como objetivo, recolher informação específica sobre um determinado momento da análise. O questionário de caracterização foi utilizado para obter informação que se caracteriza os de participantes em relação à sua experiência de utilização de tecnologias com interação por voz, bem como os seus hábitos de consumo audiovisual. O guião de observação permitiu que fosse feito o registo do conhecimento prévio dos utilizadores para utilizar as funcionalidades propostas, se durante a experiência conseguiram executar sem ajuda e qual foi o número de erros efetuados até chegar ao objetivo de cada tarefa proposta, bem como algumas observações pontuais de pequenos detalhes que foram detetados ao longo da experiência e que são mencionados ao longo deste capítulo. Ao realizar a entrevista semiestruturada foi possível obter informação sobre a facilidade de uso da interação por voz e do telecomando com o sistema MEO de uma forma mais objetiva, através do *System Usability Scale* (SUS) e de algumas questões que respondem à pergunta de investigação desta dissertação. Segundo Brooke (1995), esta escala é um método confiável e económico que pode ser aplicado em avaliações globais para medir a usabilidade de um sistema.

Dada a natureza deste tipo de entrevista, houve flexibilidade para introduzir novas questões mediante a interação com cada utilizador, que foram importantes para esclarecer alguns dos problemas que foram evidenciados por alguns utilizadores, no decorrer da experiência.

Como suporte ao método de abordagem (estudo de caso), fez-se uma gravação de vídeo e áudio de cada entrevista. Estas gravações tiveram como objetivo principal, o registo de toda a comunicação que foi feita com cada entrevistado, para recolher todos os detalhes das questões de pergunta aberta que foram colocadas no fim, mas também para guardar algum detalhe que fosse importante para analisar. Todos os participantes foram informados no início da experiência, tendo sido recolhida antecipadamente uma autorização escrita e assinada pelo entrevistado para a sua realização.

Por se tratar de um estudo que pretende avaliar a satisfação e facilidade de uso da interação por voz através linguagem natural não houve restrição de idade, pelo que o grupo foi angariado através de uma amostra de conveniência. Dado ao carácter individual da entrevista, foi permitido que cada participante pudesse dar a sua opinião sincera sobre cada questão que lhe tenha sido colocada. O investigador tem um papel fundamental para manter a conversa centrada no objetivo da entrevista, sendo o tema em estudo relativamente recente e que pode ser novidade para alguns dos entrevistados, a conversa pode dispersar com comentários ou tópicos irrelevantes para o decorrer da investigação.

2.Avaliação e validação do protótipo

Como foi descrito na secção anterior, as entrevistas permitiram avaliar a facilidade de uso entre um telecomando e interação por linguagem natural através da voz com o sistema MEO.

Para esta avaliação, foram convidados um grupo de colaboradores da Altice Labs e um grupo de pessoas externas à empresa. Para validar a arquitetura proposta nesta dissertação, este momento constou da comparação da satisfação e facilidade de uso entre o telecomando e a interação por linguagem natural por voz com o sistema MEO.

As entrevistas ocorreram nos dias 27 e 28 de setembro e houve uma participação de doze homens (80% da totalidade da amostra) e três mulheres (20% da totalidade da amostra), com idades compreendidas entre os 16 e os 67 anos.

Figura 26 - Sala Future Labs na Altice Labs

O Future Labs da Altice Labs foi o ambiente utilizado para realizar as entrevistas. É um local onde costumam decorrer apresentações de alguns dos projetos que aí estão dispostos (de *Internet of Things*, de domótica, de impressão 3D, entre outros), que foram desenvolvidos por vários colaboradores da Altice Labs. O aspeto tecnológico e inovador desta sala foi escolhido a pensar que pudesse desencadear alguma curiosidade nos entrevistados e os fizesse sentir mais motivados para a realização desta entrevista. O cuidado que foi dado na explicação de cada tarefa também encorajou uma comunicação mais informal com cada entrevistado, tendo sido fulcral para promover a interação com o sistema que foi protótipado e entender melhor como é que os entrevistados reagiam à medida que se iam sentindo mais confortáveis, ao utilizar o sistema MEO com recurso à linguagem natural por voz.

2.1. Descrição das Entrevistas semiestruturadas

No início de cada entrevista, com a duração de quarenta e cinco minutos cada, foi pedida uma autorização aos participantes para que facultassem o consentimento informado por escrito, esclarecido e livre para participarem no estudo, bem como para a captação de vídeo e som, para posterior análise dos dados obtidos. De seguida foi apresentado o contexto da dissertação, seguido

de uma explicação sobre o seguimento da entrevista. Apesar de o tempo ser uma limitação durante o momento das entrevistas, devido ao acesso condicionado do espaço, foi possível realizar as quinze entrevistas como foi previsto.

Depois de uma breve introdução, explicou-se a arquitetura do projeto, com os componentes que estavam dispostos em cima da mesa e pediu-se para preencherem o primeiro questionário, que permitiu caracterizar o participante enquanto consumidor de conteúdos televisivos e também quanto à sua experiência de utilização de tecnologias com interação por voz.

Após a finalização deste primeiro questionário, iniciou-se um grupo de questões para perceber se estes já tinham utilizado alguma vez o sistema MEO e se estavam familiarizados com as funcionalidades que iriam ser objeto do estudo (gravações manuais, gravações automáticas e fazer um favorito). Durante esta fase foram registados, numa grelha de observação, o registo das respostas dos participantes. Foi perguntado se sabiam interagir por voz com o protótipo que se pretendia validar. De seguida, mediante a resposta obtida, exemplificou-se uma demonstração de cada funcionalidade, primeiro com o telecomando e depois com voz. Aos que sabiam interagir com o telecomando, foi-lhes pedido para demostrarem como acedem. Para alguns entrevistados foi-lhes apresentado algumas alternativas mais simples. Quando chegou o momento de interagir por voz, foi-lhes apresentado várias formas de interagirem, por cada funcionalidade, para que pudessem perceber que não existe apenas um caminho certo para interagirem com o sistema. Considerou-se que desta forma permitiria que todos os utilizadores pudessem estar ao mesmo nível antes de interagirem sem ajuda, independentemente da experiência que tinham.

Durante a fase de explicação, iam surgindo notificações dos conteúdos que foram agendados com os utilizadores, tendo sido explicado o seu conceito nessa fase, referindo também uma notificação personalizada pelo sistema quando se liga a STB MEO, que foi apresentada no fim da explicação. A interface “Os meus Conteúdos” também foi explicada, mostrando o que acontece quando o utilizador interage com um conteúdo que segue e com um conteúdo que marcou para ver mais tarde.

Após a exemplificação da execução de cada funcionalidade por telecomando e por voz foi pedido a cada participante que seguisse um plano de tarefas para executar todas as funcionalidades sem ajuda, primeiro com o telecomando e depois por voz, como se pode observar na Figura 27. O plano de tarefas apresentado consistiu em:

1. Gravar um conteúdo com o telecomando;
2. Fazer um favorito de um conteúdo com o telecomando;
3. Pedir para ver este conteúdo mais tarde por voz;
4. Agendar um conteúdo por voz;
5. Pedir para seguir um conteúdo por voz;
6. Abrir a página os meus conteúdos por voz;

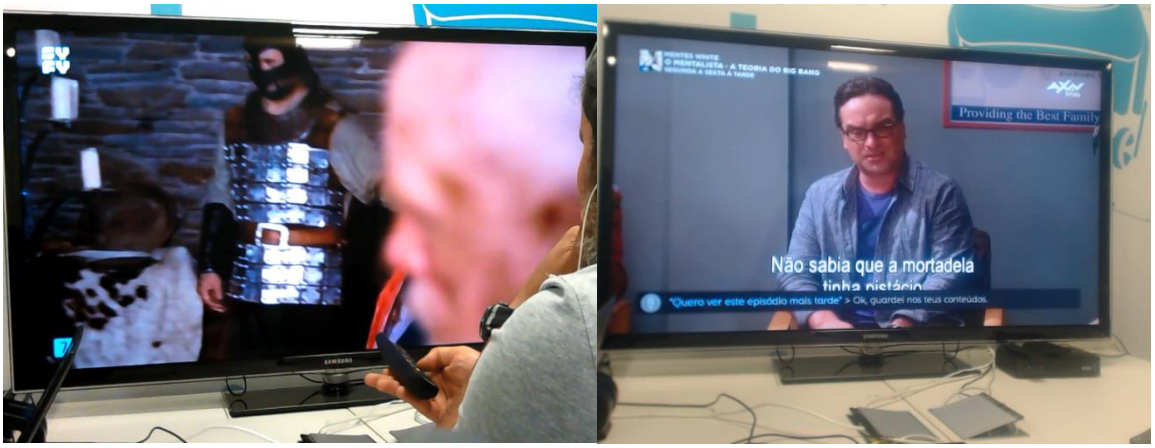


Figura 27 - Execução das tarefas gravar um conteúdo com o telecomando (esquerda) e ver mais tarde (direita)

Esta fase também foi registada na mesma grelha de observação. Deve-se salientar que este plano contempla duas funcionalidades, "Agendar" e "abrir os Meus Conteúdos" que foram desenvolvidas para interagir por voz através de linguagem natural. Como tal, não se pode aceder a estas funcionalidades com o telecomando. Quando este momento de avaliação acabou, os participantes foram incentivados a exporem a sua opinião e pensamentos. De seguida foi pedido para realizarem um questionário sobre esta fase.

Por fim, fizeram-se algumas perguntas relacionadas com a interface que dava feedback da voz e em que situações fariam uso da interação por linguagem natural através da voz.

Todos os inquiridos realizaram com sucesso todos os passos da entrevista. Após o momento de recolha de dados, procedeu-se à interpretação dos mesmos. Esta interpretação foi realizada através das várias descrições estatísticas, onde a representação poderá ser mais expressiva.

As perguntas de questão aberta que foram realizadas no fim de cada entrevista semiestruturada resultaram em dados qualitativos que foram tratados de modo a aprofundar melhor os aspetos gráficos da interface que dão *feedback* sobre o estado da interação por voz, para compreender melhor a atitude por parte dos inquiridos em relação à avaliação.

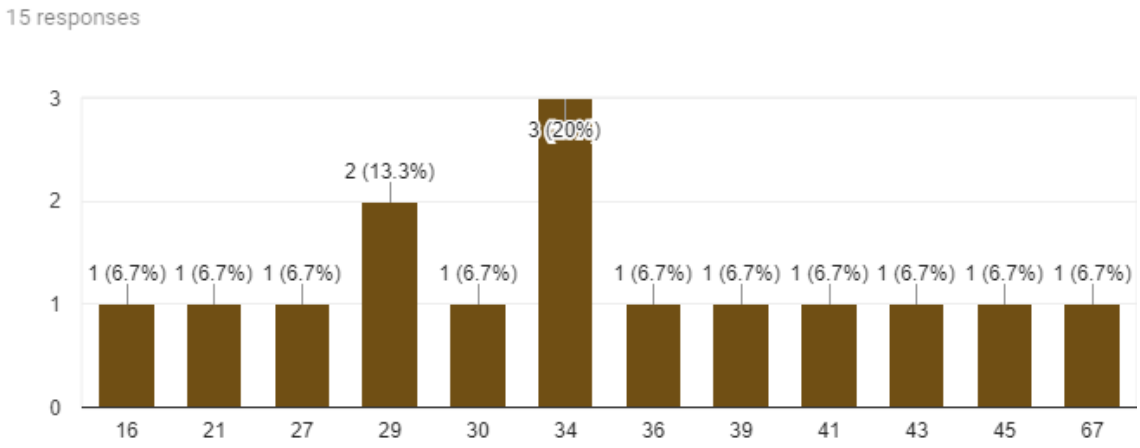
Para a análise de resultados foi utilizado o *software Microsoft Excel*, onde se mapearam todas as questões dos dois questionários, bem como do guião de observação em diferentes variáveis.

2.1.1. Observação dos resultados obtidos do Questionário de Caracterização

O questionário de caracterização pretendeu caracterizar a amostra demograficamente (ver Anexo 1), como consomem conteúdos audiovisuais e qual a experiência de utilização do inquirido com sistemas que permitem interagir por voz. Para além disso, pretendeu-se validar o interesse das funcionalidades que foram implementadas no protótipo, bem como o interesse na utilização da interação por voz através de linguagem natural. Deste questionário retiraram-se as perguntas que se consideraram pertinentes para responder à questão de investigação. Os resultados que não foram contemplados neste capítulo encontram-se nos anexos.

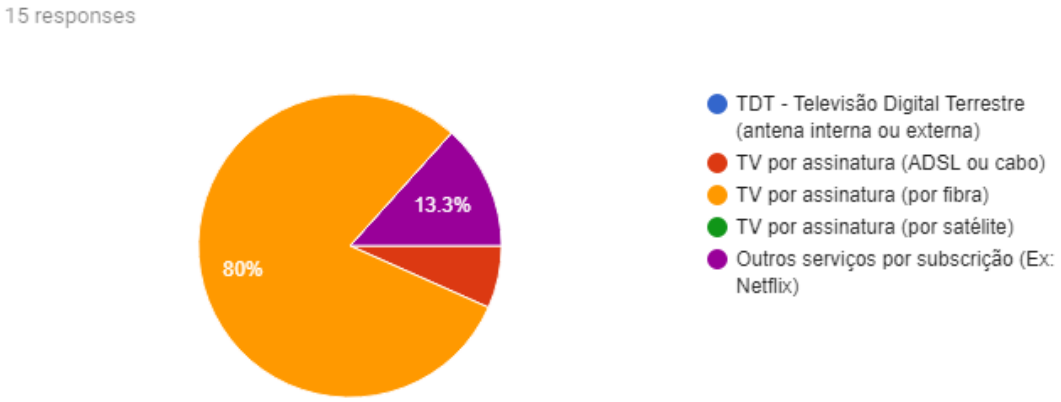
Dos quinze inquiridos que participaram na entrevista, dez são homens (66% da totalidade da amostra) e cinco são mulheres (33% da totalidade da amostra). O Gráfico 1 apresenta a distribuição dos inquiridos por idade. Como se pode observar, as idades estão compreendidas entre os 16 e os 67 anos, com predominância da faixa etária dos 30 aos 40 anos.

Gráfico 1 – Distribuição dos inquiridos por idade



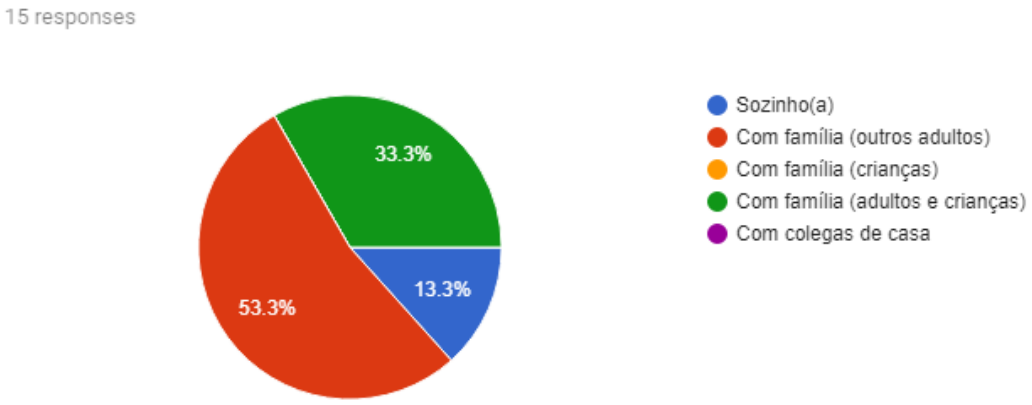
Em relação ao consumo audiovisual, o Gráfico 2 mostra que os inquiridos costumam utilizar mais a televisão como um meio de acesso a conteúdos televisivos. Apenas 2 inquiridos referiram que apenas utilizavam outros serviços de subscrição (Netflix).

Gráfico 2 - Tipos de acesso de conteúdos televisivos



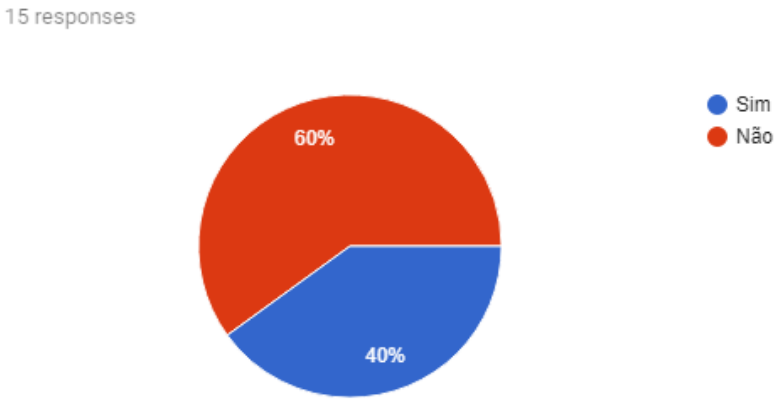
Destes inquiridos, apenas dois inquiridos vêm televisão sozinhos e os restantes com família (com adultos e adultos e crianças), como se pode observar no Gráfico 3. Este dado é cruzado ao longo do texto com a pergunta de questão aberta que foi colocada aos participantes, "como é que utilizaria mais a interação por linguagem natural, sozinho e/ou acompanhado?", para tentar perceber se a resposta se adequa com a forma como habitualmente vêm televisão.

Gráfico 3 - Visualização de televisão em casa



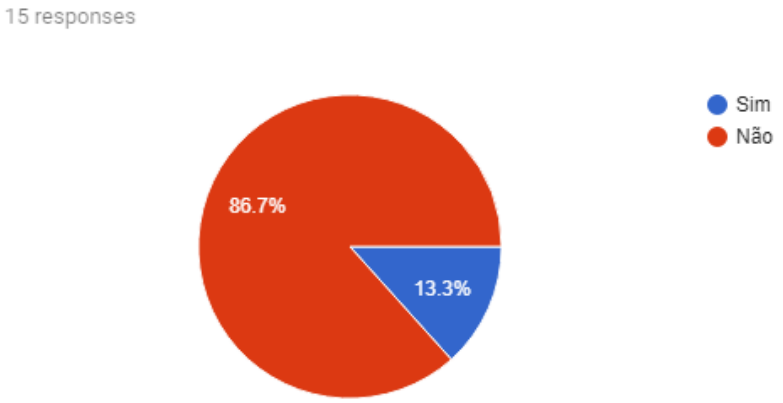
No que diz respeito às questões sobre a experiência de utilização de tecnologias com interação por linguagem natural na televisão, o Gráfico 4 mostra que 60% dos inquiridos (9 respostas) nunca interagiram com uma aplicação/sistema por voz.

Gráfico 4 - Interagiu com uma app/sistema por voz



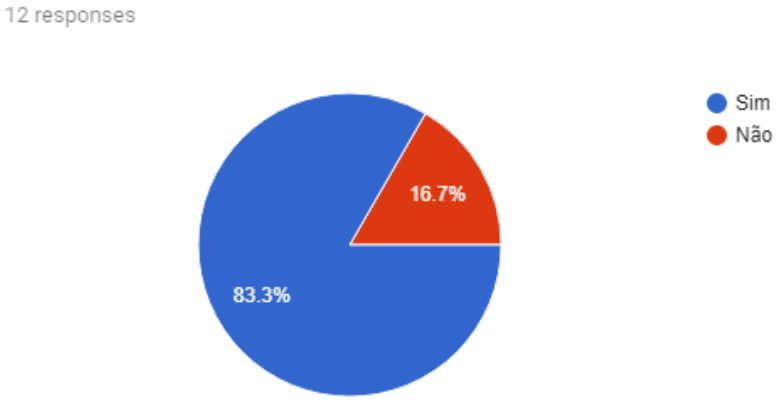
No entanto, dos seis inquiridos que já interagiram por voz, apenas dois costumam interagir através de comandos por voz com a televisão, como se pode observar no Gráfico 5.

Gráfico 5 - Interage regularmente com comandos por voz para aceder a conteúdos televisivos



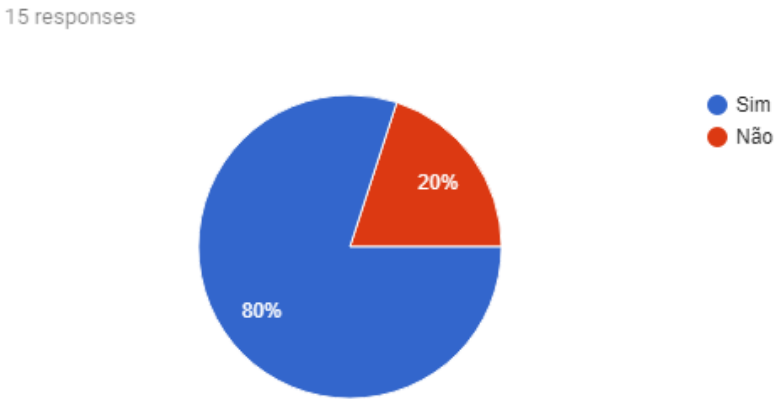
Não obstante aos resultados obtidos do Gráfico 4 e do Gráfico 5, 83,3% dos inquiridos (10 respostas) tem interesse em utilizar a linguagem natural por voz, para interagir com a televisão.

Gráfico 6 - Interesse em interagir por linguagem natural



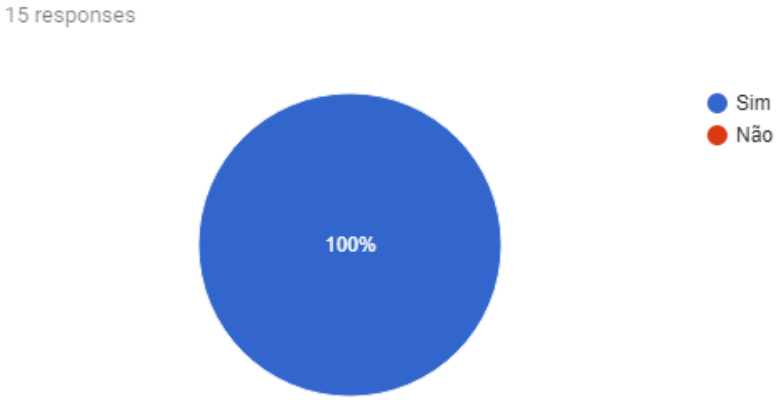
Neste questionário foi validado também, a pertinência das funcionalidades que foram implementadas com interação por linguagem natural. No Gráfico 7 podemos observar que doze inquiridos (80% da amostra total) têm interesse que o sistema lhes recomende conteúdos através de um diálogo por interação de linguagem natural por voz.

Gráfico 7 - Interesse em que o sistema seja proactivo e recomende conteúdos com interação por linguagem natural



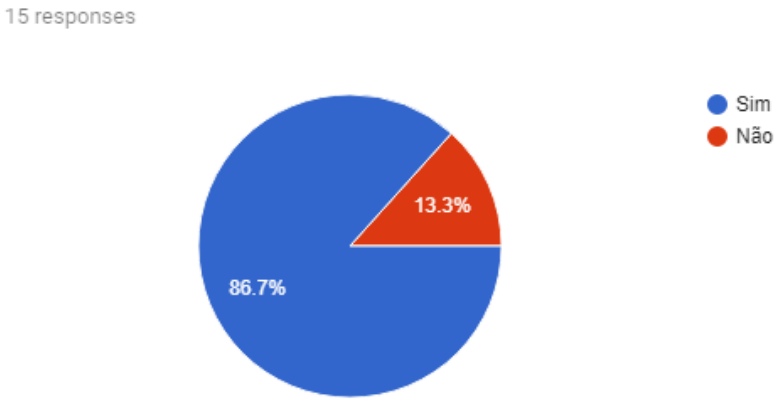
No que diz respeito ao tipo de conteúdos que o sistema recomenda, validou-se a pertinência de notificar conteúdos que o utilizador segue. Verificou-se que quinze inquiridos (100% da amostra total), têm interesse que o sistema sugira conteúdos que seguem.

Gráfico 8 - Interesse em que o sistema sugira conteúdos que segue



No que diz respeito aos conteúdos que um utilizador segue, foi inquirido qual é o interesse dos participantes ao receber notificações de conteúdos que seguem e que estão para expirar. Como podemos observar no Gráfico 9, treze inquiridos (86,7% da amostra total) têm interesse que o sistema lhes notifique quando existem conteúdos que segue e que vão expirar.

Gráfico 9 - Interesse em que o sistema sugira conteúdos que segue e que vão expirar



Através deste questionário, foi possível caracterizar a amostra e validar conceptualmente a pertinência das funcionalidades que foram consideradas em conjunto com a Altice Labs, bem como o interesse na interação por linguagem natural através da voz.

2.1.2. Observação dos resultados obtidos da interação com o protótipo

No início da parte prática da entrevista explicaram-se todas as funcionalidades que constam no plano de tarefas. Numa primeira fase, registou-se no guião de observação (ver Anexo 3) o nível de conhecimento que os participantes tinham das funcionalidades e que experiência tinham com os dois tipos de interação que iriam ser testados, interação por voz e por telecomando, como se pode verificar na Tabela 2.

Tabela 2 – Total de participantes (%) que conheciam as funcionalidades

Funcionalidade	Já conheciam (%)	Sabiam executar com telecomando (%)	Sabiam executar por voz (%)
Ver mais tarde	80	73,3	0
Agendar	0	Não existe no MEO	0
Seguir um conteúdo	33,3	20	0
Abrir os Meus Conteúdos	0	Não existe no MEO	0

A razão pela qual nenhum participante conheceu as funcionalidades “Agendar” e “Abrir os meus Conteúdos”, como foi referido no capítulo de Descrição das Entrevistas, relaciona-se com o facto destas funcionalidades não constarem na base do sistema MEO e terem sido desenvolvidas em conjunção com este protótipo para se poder interagir por voz.

A maioria dos participantes conseguiram executar com sucesso todas as funcionalidades do plano de tarefas e sem ajuda para além da explicação inicial, como se pode observar no Tabela 3.

Tabela 3 – Total de participantes que conseguiram executar as funcionalidades

Funcionalidade	Executaram com o telecomando (%)	Enganaram-se com o telecomando	Executaram por voz (%)	Enganaram-se por voz
Ver mais tarde	100	0	100	1
Agendar	Não existe no MEO	0	93,3	5
Seguir um conteúdo	100	2	100	2
Abrir os Meus Conteúdos	Não existe no MEO	0	100	2

No que diz respeito à interação por voz, houve um participante que não conseguiu executar a funcionalidade "agendar" porque se esqueceu de utilizar a *keyword* "Agendar". No entanto, outros participantes também tiveram dificuldades na execução desta tarefa, como se pode observar na coluna dos erros de interação por voz, porque ou utilizavam a *keyword* "gravar" em vez de "agendar", ou não se apercebiam que depois da primeira interação, o sistema entrava num diálogo com o participante.

O NLU não tem associada a *keyword* "gravar" para esta funcionalidade, porque estas as *keywords* têm significados diferentes, podendo causar confusão no utilizador, com uma das funcionalidades que existe no MEO, a gravação literal de um conteúdo no disco rígido da STB. Quando se questionaram os participantes o porquê de não perceberem que havia continuação no diálogo para agendar um conteúdo, a maioria referiu que a voz que ouvia nos auscultadores os induzia em erro, por acabar a frase na afirmativa. Apesar do texto que é enviado para a Polly conter o ponto de interrogação, esta não consegue dar a entoação adequada à frase, de modo a ser entendida como uma pergunta. Para além destes problemas, encontrou-se outro erro ao nível do NLU, quando a variável predefinida pela Microsoft que é responsável por converter texto num formato de data e hora, não interpreta "meio dia" como uma hora válida.

Na funcionalidade de "Seguir um conteúdo", os participantes enganaram-se duas vezes por utilizarem a palavra "favorito" em vez de seguir. Isto deve-se ao facto de que, embora tenha sido explicado inicialmente que o favorito do MEO (que engloba canais, programas, apps e videoclube) é diferente de seguir uma série, alguns utilizadores fizeram confusão ou esqueceram-se. No entanto, apesar deste caso se ter repetido uma terceira vez, o NLU compreendeu que o pedido era semelhante ao padrão (*pattern*) que foi definido para "seguir", tendo sido executado com sucesso.

Os erros que se surgiram da funcionalidade de "ver mais tarde" e "Abrir os Meus Conteúdos" são por má colocação de voz por parte do utilizador e/ou má captação do microfone *far-field*. Dois dos participantes não estavam a conseguir ativar a *keyword* "Sofia". Como o microfone utilizado tem uma distância ótima para ser utilizado, por vezes os utilizadores tinham de se aproximar e repetir o que disseram. Depois de se incentivar que falassem mais alto não tiveram mais problemas relacionados com este tipo.

2.1.3. Observação dos resultados obtidos do Questionário validação do protótipo

O questionário de validação e avaliação do protótipo (ver Anexo 2) permitiu comparar a satisfação e facilidade de uso do sistema MEO entre o protótipo de interação por voz e o telecomando através da administração do *System Usability Scale* (SUS). Depois de se calcular as pontuações finais do SUS para cada participante, como Brooke (1995) refere e calcular a média total, as pontuações finais para os dois tipos de interação podem ser observados na Tabela 4.

Tabela 4 - Pontuação SUS final dos dois tipos de interação avaliados

Tipo de Interação	Pontuação SUS
Telecomando	70,7
Voz através de linguagem natural	77

Os resultados obtidos da pontuação SUS da interação com o telecomando são similares aos resultados da interação por voz. No entanto, através da Figura 28, é possível observar que o resultado obtido da interação com o telecomando está no início do intervalo de acessibilidade aceitável (*Acceptable*), enquanto que a interação por voz ficou bem estabelecida no centro desse intervalo, a três pontos de ser considerado o grau B. Desta forma, segundo a pontuação através dos adjetivos, pode-se considerar que a interação por voz é satisfatória (ok) e a interação por voz é boa (*good*).

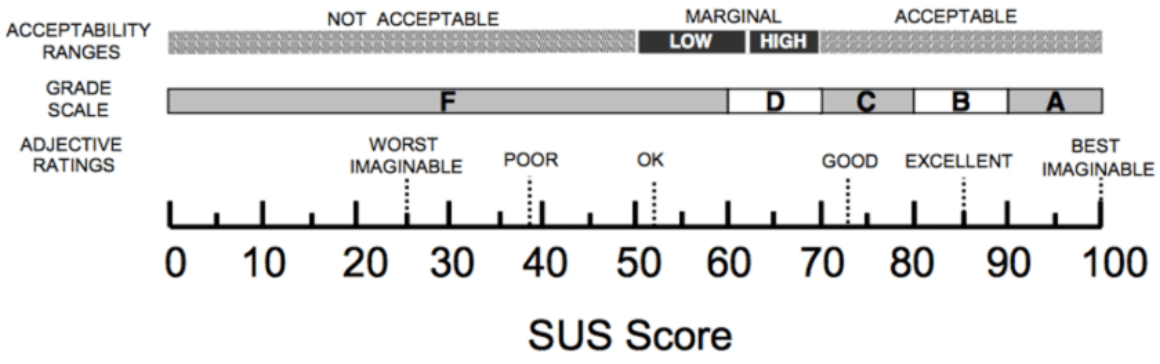


Figura 28 - Pontuação do SUS (Bangor, Kortum, & Miller, 2009)

2.1.4. Observação dos resultados obtidos da análise das perguntas

Na tabela seguinte, é apresentado de forma sumária as respostas dos participantes, que serão analisadas ao longo deste capítulo.

Tabela 5 - Análise das respostas dos Inquiridos

Perguntas	Respostas
O que é que achou do feedback visual que obteve?	- "a cor do texto e do ícone era percetível" - "acho que estava adequado" - "Por vezes é um pouco lento"

	- “está bem integrado no sistema” - “para um daltónico a cor é um pouco subtil, mas percebe-se” - “não achei que seria disruptivo àquilo que estava a ver na televisão”
Sente que o diálogo não precisa de feedback, ou que é uma mais valia?	**Sim**. - “sim, porque consegues ver o que estás a dizer” - “sim, principalmente por a voz do sistema acabar sempre de forma afirmativa” - “gosto de saber o que o sistema percebeu”
E quando o sistema não percebe exatamente o que diz?	- “é frustrante, mas ele até percebe bem” - “dá para perceber que o sistema não me entendeu” - “não achei muito complicado” - “é preciso ter calma e repetir novamente”
O que achou do tempo que demora desde que fala até obter resposta?	- “acho que demora mais quando o comando está errado” - “estava muito rápido, cerca de 2 segundos” - “para interagir mais naturalmente devia demorar menos um segundo” - “pode ter demorado a dar resposta ao meu pedido, mas acho que não é nada fora do normal”
O que acha da naturalidade com que o sistema o entendeu?	**Sim**. - “percebe bastante bem o que eu quero dizer” - “feedback quase imediato” - “acho que está bem, mas ainda pode melhorar” - “não se enganou em nenhuma palavra” - “quando chamo a Sofia, gostava que ela respondesse com sim”
Acha que a perceção do sistema se aproxima da perceção de um ser humano?	- “dentro das suas limitações” - “discurso fluído e percebe bem” - “Ainda falta algum avanço, a fala é muito complexa”
Sente que consegue falar naturalmente para interagir com este sistema?	**Sim**. - “não tem uma voz parecida com o GPS” - “a voz falha na pontuação final da frase, mas não é nada robótica” - “penso que foi mais ou menos natural” - “não é a mesma coisa, mas estamos a chegar perto” - “Perfeitamente”
Como é que perspetiva que utilizaria este sistema na sua casa, numa situação em que está sozinho versus acompanhado?	- “Acompanhado era capaz de ter mais ruído” - “vejo-me utilizar este sistema quando estou a fazer tarefas em casa”

	- "se estiver acompanhado e a conversar, prefiro usar o comando" - "sozinho utiliza-se cinco estrelas" - "posso usar para colocar desenhos animados ao meu filho enquanto estou ocupado" - "também utilizaria quando estivesse acompanhado, é mais fácil interagir por voz"
Acha que o sistema poderá entrar em conflito se houver muito ruído, quando está acompanhado?	**Sim**. - "há a possibilidade de ativar sem ser necessário" - "pode não ativar quando é chamado"
Se o teu sistema permitisse, substituiria completamente a interação do telecomando pela voz?	- "Para tudo não, apenas para as funcionalidades que mais utilizo"

Estas perguntas permitiram compreender vários aspetos da aplicação, nomeadamente lacunas existentes bem como aspetos positivos que surgiram durante a interação com o protótipo. Os intervenientes referiram inúmeras vezes a aplicabilidade que este tipo interação com a televisão pode ter no seu quotidiano, principalmente quando estão a realizar tarefas que não lhes permite interagir com o telecomando. A interface visual que serve de suporte à interação por voz também foi considerada um elemento importante e bem enquadrado com o sistema MEO, principalmente para validar o que tinha sido dito e detetar erros de interação com mais facilidade. O *feedback* sobre a naturalidade com que o sistema entendeu os inquiridos foi maioritariamente positivo. No entanto, vários participantes mencionaram que a voz, apesar de não parecer robotizada, ficou aquém do que esperavam, por não fazer o uso adequado da pontuação no final da frase, induzindo-os em erro. Devido a este problema, só se aperceberam que havia continuação dos diálogos, quando verificavam o feedback visual do sistema na televisão. Quanto aos tempos de resposta, não houve um consenso entre os inquiridos, desde mencionarem que o tempo de resposta do sistema é "quase imediato", até referirem que "para interagir mais naturalmente devia demorar menos um segundo". Quase todos os inquiridos concordaram que o ruído é o principal problema ao interagir com voz, ou porque o sistema pode ser ativado sem ser necessário, ou não porque não entende corretamente o que é dito. Os participantes estão mais predispostos a utilizar a interação por voz quando estão sozinhos, porque quando estão acompanhados não querem quebrar a conversa para interagir com o sistema. No entanto, alguns mencionaram que interagiam por voz, mesmo estando acompanhados, por ser mais fácil de usar do que com o telecomando.

V. Discussão dos resultados

1. Satisfação e Facilidade de uso

Os indicadores de usabilidade utilizados apresentam um valor positivo em termos de satisfação e facilidade de uso dos utilizadores. A interação com o telecomando obteve um resultado de 70,7 no SUS, enquanto a interação por voz através de linguagem natural obteve 77 pontos, numa escala de 0 a 100. Estes indicadores mostram como o protótipo de interação por voz foi considerado com boa

usabilidade pelos 15 participantes da pesquisa. Infere-se que a satisfação dos utilizadores não foi determinada pelo desempenho satisfatório ao realizarem a parte prática da entrevista, mas sim pela facilidade de acesso aos conteúdos através da interação por voz e pelas possibilidades que existem para executar a mesma funcionalidade, necessitando apenas de recorrer à palavra chave da ação que querem executar.

2.A interface da Interação por voz

A aplicação que foi desenvolvida para dar suporte à interação por voz através de feedback visual, comprovou ser uma mais valia para a interação, principalmente por acompanhar o estado do diálogo e permitir que o utilizador possa detetar erros de interpretação do sistema. Sobre o a dimensão que a interface ocupa na televisão, tanto a caixa de texto, como as notificações que aparecem no ecrã, os utilizadores consideraram que não eram disruptivas para o que estava a ser emitido na televisão, ainda que por vezes a caixa de texto se sobrepunha às legendas. Não obstante, os utilizadores não deram muita importância a esse aspeto.

3.Pontos fortes e fracos da interação

Com base nas entrevistas e nas observações da interação com o protótipo, destacou-se como um ponto forte, a facilidade de uso da interação por voz e a aplicabilidade que este tipo interação no quotidiano, permitindo que o utilizador possa estar ocupado e interagir rapidamente com o seu sistema televisivo. A notificação que aparece quando se liga a STB também foi mencionada como uma solução pertinente por permitir que os utilizadores possam continuar a ver um conteúdo que não acabaram e por sugerir conteúdos que seguem e vão expirar. Para além disso, a implementação de *patterns* permitiu que os pedidos que inicialmente não foram entendidos pelo sistema, pudessem ser considerados no respetivo *intent,* através da comparação de palavras semelhantes à frase padrão.

No que diz respeito aos pontos fracos da interação, destacaram-se o tempo de resposta entre o pedido do utilizador e o feedback visual da ação e também, o text-to-*speech* utilizado para conceder voz ao sistema, por não conseguir dar entoação a uma frase quando é do tipo interrogativo. Uma solução apontada para o tempo de resposta seria reduzir o intervalo de tempo desde que o utilizador acabou de falar até o sistema executar a ação. Sobre o problema de interação referente ao *text-to-speech* o serviço Polly que é utilizado neste protótipo é, segundo o estudo feito pela Altice Labs, o melhor serviço em português de Portugal. Como tal, deve-se optar por outras estratégias, utilizando frases afirmativas como “Vou agendar para as 10 horas. Confirma por favor.”, para garantir que o utilizador consiga perceber que dar uma resposta.

Apresentou-se ainda como ponto fraco, o facto que que o desempenho da interação por voz estar intimamente ligado com o ruído existente na sala, o que faz com que este tipo de interação tenha menos interesse quando existem mais do que duas pessoas a ver televisão, ou existe ruído o suficiente para o sistema não conseguir perceber bem o que o utilizador disse.

VI.Conclusões

Este projeto teve como objetivo principal a conceção, construção e implementação de um protótipo de um sistema conversacional através da interação por linguagem natural, que permita interagir por voz para pedir para voltar a ver conteúdos televisivos que, já tendo sido iniciados pelo utilizador, não tenham sido finalizado, sendo que numa primeira fase foi necessário definir as funcionalidades pertinentes no desenvolvimento para depois treinar o sistema, mais concretamente a componente de Natural Language Understanding, para compreender os comandos neste contexto específico da televisão. Estas funcionalidades também são a chave para que o sistema possa recomendar conteúdos aos utilizadores, por meio de uma notificação na televisão.

Posteriormente, procedeu-se à integração das aplicações que compõem a interação por voz com o desenvolvimento de uma aplicação para televisão em Mediaroom PF, que dá suporte visual à interação por voz. Destaca-se o componente de NLU como o ingrediente principal para a existência da interação por linguagem natura, assim como a integração dos serviços de armazenamento em *cloud,* cujos dados foram utilizados pela Proactive API para lançar notificações para o sistema MEO.

Neste sentido, o presente estudo assenta-se numa metodologia de investigação de desenvolvimento com o objetivo de desenvolver um protótipo de uma aplicação de televisão que permita interagir por voz, através de linguagem natural, com o sistema MEO, com funcionalidades que sejam pertinentes para os consumidores de conteúdos televisivos.

1. Limitações do protótipo

Ao longo desta investigação foram surgindo alguns obstáculos na realização das diferentes tarefas. Uma das limitações do protótipo diz respeito à interação por voz, uma vez que não é realizada a filtragem da resposta por voz do sistema, bem como do ruído de fundo da televisão e como tal, o microfone capta os sons do ambiente como se fosse a voz do utilizador.

2. Trabalho futuro

Para dar continuidade a este projeto seria desejável a implementação e desenvolvimento da plataforma, através da criação de um Script que automatizasse o arranque de todas as apps do Raspberry Pi, sempre que este se inicia. Para além disso, o desenvolvimento de um sistema de voz é um processo contínuo, pelo que é encorajado a adição continua de mais interações diferentes para melhorar a robustez dos diálogos existentes e abranger novas funcionalidades para expandir o alcance de funcionalidades disponíveis por voz.

VII.Bibliografia

1980's Julie by Worlds of Wonder Commercial - YouTube. (2006). Retrieved December 23, 2017, from https://www.youtube.com/watch?v=UkU9SbIictc

Alan Bryman. (2012). *Social Research Methods* (4th Editio). Retrieved from https://www.academia.edu/16577475/Alan_Bryman-Social_Research_Methods_4th_Edition-Oxford_University_Press_2012_

Amazon. (2018). Amazon Alexa. Retrieved from https://developer.amazon.com/alexa

Bangor, A., Kortum, P., & Miller, J. (2009). Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Studies*, *4*(3), 114–123. Retrieved from http://dl.acm.org/citation.cfm?id=2835587.2835589

Barraza, R., & Zhang, R. (2016). Creating Speech-Driven Applications with Cognitive Services and LUIS - Developer Blog. Retrieved December 14, 2017, from https://www.microsoft.com/developerblog/2016/11/01/creating-speech-driven-applications-with-cognitive-services-and-luis/

Berglund, A., & Johansson, P. (2004). Using speech and dialogue for interactive TV navigation. *Universal Access in the Information Society*, *3*(3–4), 224–238. https://doi.org/10.1007/s10209-004-0106-x

Brooke, J. (1995). *SUS: A quick and dirty usability scale*. *Usability Eval. Ind.* (Vol. 189). Retrieved from https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale

Cohen, D., & Crabtree, B. (2006). Semi-structured Interviews Recording Semi-Structured interviews. *Qualitative Research Guidelines Project*, 2. Retrieved from https://www.sswm.info/sites/default/files/reference_attachments/COHEN 2006 Semistructured Interview.pdf

Corden, J. (2017). A brief history of Cortana, Microsoft's trusty digital assistant | Windows Central. Retrieved January 2, 2018, from https://www.windowscentral.com/history-cortana-microsofts-digital-assistant

Coutinho, C. P. (2011). *Metodologia de Investigação em Ciências Sociais e Humanas*. (S. A. Edições Almedina, Ed.) (2º Reimpre). Coimbra: GRUPOALMEDINA.

Covington, M. A. (1994). *Natural language processing for Prolog programmers*. Prentice Hall. Retrieved from https://books.google.pt/books/about/Natural_Language_Processing_for_Prolog_P.html?id=mX736RH9VdUC&redir_esc=y

Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T. (1987). The vocabulary problem in human-system communication. *Communications of the ACM*, *30*(11), 964–971. https://doi.org/10.1145/32206.32212

Gazdar, G., & Mellish, C. S. (Christopher S. . (1989). *Natural language processing in Prolog : an introduction to computational linguistics*. Addison-Wesley Pub. Co. Retrieved from https://books.google.pt/books?id=1fUYAQAAIAAJ&q=Natural+language+processing+in+Prolog&dq=Natural+language+processing+in+Prolog&hl=en&sa=X&ved=0ahUKEwi7iP3TueDYAhVNxKYKHf85DTcQ6AEIOTAD

Get started with the Microsoft Speech Recognition API by using the C# desktop library | Microsoft Docs. (2017). Retrieved January 5, 2018, from https://docs.microsoft.com/en-us/azure/cognitive-services/speech/getstarted/getstartedcsharpdesktop

Google. (2017). Speech API - Reconhecimento de fala | Google Cloud Platform. Retrieved December 14, 2017, from https://cloud.google.com/speech/?utm_source=google&utm_medium=cpc&utm_campaign=emea-emea-all-en-dr-skws-all-all-trial-e-gcp-1002258&utm_content=text-ad-none-any-DEV_c-CRE_219695715317-

ADGP_SKWS+%7C+EXA+~+M%3A1_PT_EN_ML_Speech+API_cloud+speech+api-KWID_4

Gorelick, C. (2017). TV is the OG of user-centered design – Design Voices – Medium. Retrieved January 3, 2018, from https://medium.com/design-voices/the-og-of-user-centered-design-tv-3e0aca65f3f0

Healey, B. (2016). NLP vs. NLU: What's the Difference? – Lola – Medium. Retrieved December 14, 2017, from https://medium.com/@lolatravel/nlp-vs-nlu-whats-the-difference-d91c06780992

Hope, C. (2017). When was the first computer invented? Retrieved December 23, 2017, from https://www.computerhope.com/issues/ch000984.htm

IBM. (2017). Watson Speech to Text. Retrieved December 14, 2017, from https://www.ibm.com/watson/services/speech-to-text/

IBM Watson. (2017). Retrieved December 12, 2017, from https://www.ibm.com/watson/about/index.html

Johansson, P. (2003). Natural Language Interaction in Personalized EPGs. *Proceedings of the 3rd UM Workshop ``Personalization in Future TV''*, 27–31.

Juang, B. H., & Rabiner, L. R. (2004). Automatic Speech Recognition – A Brief History of the Technology Development. Retrieved from http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/354_LALI-ASRHistory-final-10-8.pdf

Katzmaier, D. (2017). Google Assistant voice control coming to Android TV - CNET. Retrieved November 26, 2017, from https://www.cnet.com/news/google-assistant-voice-control-coming-to-android-tv-this-summer/

KITT.AI NLU. (2018). Intents and Entities — KITT.AI NLU 0.1 documentation.

Lafferty, M. (2016). Designing for Television, Part 1 – This Also – Medium. Retrieved

January 3, 2018, from https://medium.com/this-also/designing-for-television-part-1-54508432830f

Lemmetty, S. (2017). Problems in Speech Synthesis. Retrieved January 10, 2018, from http://research.spa.aalto.fi/publications/theses/lemmetty_mst/chap4.html

Maria Alves de Oliveira, K., Paola Aguiar, Y., Júnior Lula, B., Carlos Rodrigues Chaves, Luiz Guedes, G., Vieira, D. A., Ygor, O. C., … Alves, M. de O. (2007). *O Uso de modelos e Múltiplos Protótipos na Concepção de Interface do Usuário*. Retrieved from http://periodicos.ifpb.edu.br/index.php/principia/article/viewFile/258/216

Microsoft. (2017). Bing Speech API - Speech Recognition Software | Microsoft Azure. Retrieved December 16, 2017, from https://azure.microsoft.com/en-us/services/cognitive-services/speech/

Norman, D. A., & Draper, S. W. (1986). *User centered system design : new perspectives on human-computer interaction*. L. Erlbaum Associates.

Nuance. (2017). Dragon speech recognition overview. Retrieved December 14, 2017, from http://shop.nuance.co.uk/store/nuanceeu/en_GB/Content/pbPage.microsite-dragon-overview?currency=EUR&pgmid=95409400&utm_source=google&utm_medium=emea-cpc&utm_campaign=DBU+%2F+DNS+Speech+Recognition+%2F+Generic+%2F+EU+-+EN+%2F+EN+%2F+None+%2F+Broad&keyword=

Oliveira, L. R. (2006). Metodologia do desenvolvimento : um estudo de criação de um ambiente de e-learning para o ensino presencial universitário. *Educação Unisinos*, *10*(1), 69–77. Retrieved from https://cld.pt/dl/download/f6828625-c3e3-40ea-9b2e-4bdc13c5c44f/MetDesenvolvimento.pdf

Pinola, M. (2011). History of voice recognition: from Audrey to Siri. Retrieved November 14, 2017, from https://www.itbusiness.ca/news/history-of-voice-recognition-from-audrey-to-siri/15008

Rabiner, L., & Juang, B.-H. (2008). Historical Perspective of the Field of ASR/NLU. In *Springer Handbook of Speech Processing* (pp. 521–538). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-49127-9_26

Richey, R. C., & Klein, J. D. (2005). Developmental Research Methods: Creating Knowledge from Instructional Design and Development Practice. *Journal of Computing in Higher Education Spring*, *16*(2), 23–38. Retrieved from https://cld.pt/dl/download/a5e931b1-637d-49b1-b018-e7b153ad605e/Richey_Klein_2005.pdf

Slavetskiy, D. (2016). Talking to machines more naturally than ever before—voice interface for Lekta NLP – LEKTA BLOG. Retrieved February 2, 2018, from https://lekta.ai/blog/talking-to-machines-more-naturally-than-ever-before-voice-interface-for-lekta-nlp/

Van der Maren, J.-M. (1996). *Méthodes de recherche pour l'éducation*. De Boeck Université. Retrieved from https://papyrus.bib.umontreal.ca/xmlui/handle/1866/4688

Weinberger, M. (2017). Amazon Echo and Alexa history: From speaker to smart home hub - Business Insider. Retrieved January 2, 2018, from http://www.businessinsider.com/amazon-echo-and-alexa-history-from-speaker-to-smart-home-hub-2017-5

Whitenton, K. (2017). Audio Signifiers for Voice Interaction. Retrieved February 2, 2018, from https://www.nngroup.com/articles/audio-signifiers-voice-interaction/?utm_source=Alertbox&utm_campaign=0741ff983b-audiosignifiers_dontvalidatedesign_2017_09_11&utm_medium=email&utm_term=0_7f29a2b335-0741ff983b-24092741

Williams, J., Raux, A., Ramachandran, D., & Black, A. (2013). The dialog state tracking challenge. In *14th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL 2013* (pp. 404–413). Microsoft Research, Redmond, WA, United

States: Association for Computational Linguistics (ACL). Retrieved from https://www.scopus.com/inward/record.uri?eid=2-s2.0-84987896853&partnerID=40&md5=c87e489a24125b25f475d6e1443e7893

Williams, R. (2015). Apple TV: Hands on review of Apple's set-top television box - Telegraph. Retrieved December 31, 2017, from http://www.telegraph.co.uk/technology/apple/11855344/Apple-TV-Hands-on-with-Apples-new-television-box.html

Yin, R. K. (2001). *Estudo de Caso: planejamento e métodos*. (Bookmam, Ed.) (2ª Ed.). Porto Alegre. Retrieved from https://saudeglobaldotorg1.files.wordpress.com/2014/02/yin-metodologia_da_pesquisa_estudo_de_caso_yin.pdf

Zajechowski, M. (2014). Automatic Speech Recognition (ASR) Software - An Introduction - Usability Geek. Retrieved December 13, 2017, from https://usabilitygeek.com/automatic-speech-recognition-asr-software-an-introduction/

Zibreg, C. (2017). How Apple created Siri's personality from Susan Bennett's original voice work. Retrieved January 2, 2018, from http://www.idownloadblog.com/2017/04/14/how-apple-created-siris-personality-from-susan-bennetts-original-voice-work/

Anexos

Anexo 1 - Questionário de caracterização sociodemográfica, consumo audiovisual e experiência de interação por voz

Caracterização de consumo audiovisual

Dissertação de Mestrado

* Required

1 - Caracterização sociodemográfica

1. **1.1 - Género** *
Mark only one oval.
- ◯ Masculino
- ◯ Feminino

2. **1.2 - Idade** *

_____

3. **1.3 - Habilitações Literárias** *
Mark only one oval.
- ◯ Ensino Básico
- ◯ Ensino Secundário
- ◯ Curso profissional
- ◯ Bacharelato
- ◯ Licenciatura
- ◯ Pós-graduação
- ◯ Mestrado
- ◯ Doutoramento

4. **1.4 - Situação Profissional Atual** *
Mark only one oval.
- ◯ Estudante
- ◯ Trabalhador
- ◯ Trabalhador-Estudante
- ◯ Desempregado
- ◯ Reformado

5. **1.5 - Quando está em casa como costuma, maioritariamente, ver televisão** *
Mark only one oval.

- ◯ Sozinho(a)
- ◯ Com família (outros adultos)
- ◯ Com família (crianças)
- ◯ Com família (adultos e crianças)
- ◯ Com colegas de casa
- ◯ Other: _____

Caracterização de consumo audiovisual

2 - Acesso a conteúdos e consumo televisivo

6. **2.1 - Indique que tipo de acesso a conteúdos televisivos (direto e/ou diferido) possui:** *
Mark only one oval.

- ◯ TDT - Televisão Digital Terrestre (antena interna ou externa)
- ◯ TV por assinatura (ADSL ou cabo)
- ◯ TV por assinatura (por fibra)
- ◯ TV por assinatura (por satélite)
- ◯ Outros serviços por subscrição (Ex: Netflix)
- ◯ Other: _____

7. **2.2 Indique com que frequência utiliza cada uma das seguintes funcionalidades/serviços na sua TV:** *
Mark only one oval per row.

	Não tenho	Não tenho, mas gostaria de ter	Tenho, mas não utilizo	Pelo menos uma vez por dia	Menos de 1 vez por semana	Algumas vezes por semana
Videoclube	◯	◯	◯	◯	◯	◯
Gravações automáticas	◯	◯	◯	◯	◯	◯
Continuar a Ver	◯	◯	◯	◯	◯	◯

8. **2.3 - Indique os dias em que, habitualmente, vê TV em casa:** *
Check all that apply.

- ☐ Todos os dias
- ☐ Segunda-feira
- ☐ Terça-feira
- ☐ Quarta-feira
- ☐ Quinta-feira
- ☐ Sexta-feira
- ☐ Sábado
- ☐ Domingo

9. **2.4 - Indique com que frequência assiste cada uma das seguintes categorias de programas televisivos:** *
Mark only one oval per row.

	Não vejo	Raramente	Algumas vezes por dia	Menos de 1 vez por semana	Algumas vezes por semana	Todos os dias da semana
Filmes						
Séries						
Programas de Entretenimento (Reality Shows, Talent Shows, Programas da manhã/tarde, Concursos, Telenovelas...)						
Documentários						
Programas de Desporto						
Programas de Informação e Telejornais						

10. **2.5 - Indique uma estimativa do tempo diário que despende a ver televisão, em casa, nos dias úteis:** *
Mark only one oval.

- 0H00m
- 0H30m
- 1H00m
- 1H30m
- 2H00m
- 2H30m
- 3H00m
- 3H30m
- 4H00m
- 4H30m
- 5 Horas ou mais

11. **2.5 - Indique uma estimativa do tempo diário que despende a ver televisão, em casa, no fim-de-semana:** *
Mark only one oval.

- ◯ 0H00m
- ◯ 0H30m
- ◯ 1H00m
- ◯ 1H30m
- ◯ 2H00m
- ◯ 2H30m
- ◯ 3H00m
- ◯ 3H30m
- ◯ 4H00m
- ◯ 4H30m
- ◯ 5 Horas ou mais

Caracterização de consumo audiovisual

3 - Iteração por voz

12. **3.1 - Já alguma vez interagiu com uma aplicação/sistema por voz?** *
Mark only one oval.

- ◯ Sim
- ◯ Não

13. **3.2 - No caso de interagir por voz com aplicações/sistemas, que assistente(s) utiliza?**
Check all that apply.

- ☐ Alexa
- ☐ Siri
- ☐ Google
- ☐ Bing Voice
- ☐ Watson
- ☐ Nuance
- ☐ Bixby

14. **3.3 - Costuma interagir com comandos por voz para aceder a conteúdos televisivos?** *
Mark only one oval.

- ◯ Sim
- ◯ Não

15. **3.4 - No caso de interagir por voz com a sua televisão, que assistente(s) utiliza na sua casa?**
Check all that apply.

- [ ] Alexa
- [ ] Siri
- [ ] Google
- [ ] Bing Voice
- [ ] Watson
- [ ] Nuance
- [ ] Bixby

16. **3.5 - Costuma interagir com a televisão através de linguagem natural por voz?** *
Mark only one oval.

- ◯ Sim
- ◯ Não

17. **3.6 - No caso de não interagir com a televisão através de linguagem natural por voz, gostaria de utilizar se tivesse oportunidade?**
Mark only one oval.

- ◯ Sim
- ◯ Não

18. **3.7 - Gostaria que o seu sistema interagisse consigo por linguagem natural de forma proativa, para lhe recomendar conteúdos?**
Mark only one oval.

- ◯ Sim
- ◯ Não

19. **3.8 - Se respondeu negativamente na alínea anterior, justifique porquê.**

_____

20. **3.9 - Gostaria que o seu sistema lhe sugerisse quando existem novos episódios de conteúdos que costuma seguir?**
Mark only one oval.

- ◯ Sim
- ◯ Não

21. **3.10 - Gostaria que o seu sistema lhe sugerisse quando existem episódios de conteúdos que segue e estão prestes a expirar do seu sistema televisivo?**
Mark only one oval.

- ◯ Sim
- ◯ Não

Anexo 2 - Questionário de validação e avaliação do protótipo

1-Avaliação do protótipo

1. **1.1- Depois de interagir com o protótipo, gostaria de utilizar estas funcionalidades de interacção por voz no seu sistema televisivo?** *
   Mark only one oval.
   - ◯ Sim
   - ◯ Não
   - ◯ Talvez

2. **1.2 - Se a interação por linguagem natural permitisse interagir totalmente com a sua televisão, prescindia do seu telecomando como forma de interação principal?** *
   Mark only one oval.
   - ◯ Sim
   - ◯ Não
   - ◯ Talvez

3. **1.3 - Diga de 1 a 5 (sendo 1 muito negativo e 5 muito positivo) se considera que interagir por linguagem natural facilitou a interacção com a televisão.** *
   Mark only one oval.

	1	2	3	4	5	
Muito Negativo	◯	◯	◯	◯	◯	Muito Positivo

4. **1.4 - Diga de 1 a 5 (sendo 1 muito negativo e 5 muito positivo) se considera que a interacção por linguagem natural facilita melhor o acesso aos tipos de conteúdos apresentados, do que com o telecomando.** *
   Mark only one oval.

	1	2	3	4	5	
Muito Negativo	◯	◯	◯	◯	◯	Muito Positivo

Avaliação da Usabilidade do produto com o Telecomando

Questionário SUS (System Usability Scale)

5. **1- Acho que gostaria de utilizar este produto com frequência.** *
   Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

6. **2- Considerei o produto mais complexo do que necessário.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

7. **3- Achei o produto fácil de utilizar.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

8. **4- Acho que necessitaria de ajuda de um técnico para conseguir utilizar este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

9. **5- Considerei que as várias funcionalidades deste produto estavam bem integradas.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

10. **6- Achei que este produto tinha muitas inconsistências.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

11. **7- Suponho que a maioria das pessoas aprenderia a utilizar rapidamente este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

12. **8- Considerei o produto muito complicado de utilizar.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

13. **9- Senti-me muito confiante a utilizar este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

14. **10- Tive que aprender muito antes de conseguir lidar com este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

Avaliação da Usabilidade do produto com voz

Questionário SUS (System Usability Scale)

15. **1- Acho que gostaria de utilizar este produto com frequência.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

16. **2- Considerei o produto mais complexo do que necessário.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

17. **3- Achei o produto fácil de utilizar.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

18. **4- Acho que necessitaria de ajuda de um técnico para conseguir utilizar este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

19. **5- Considerei que as várias funcionalidades deste produto estavam bem integradas.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

20. **6- Achei que este produto tinha muitas inconsistências.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

21. **7- Suponho que a maioria das pessoas aprenderia a utilizar rapidamente este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

22. **8- Considerei o produto muito complicado de utilizar.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

23. **9- Senti-me muito confiante a utilizar este produto.** *
Mark only one oval.

	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

24. **10- Tive que aprender muito antes de conseguir lidar com este produto.** *
Mark only one oval.

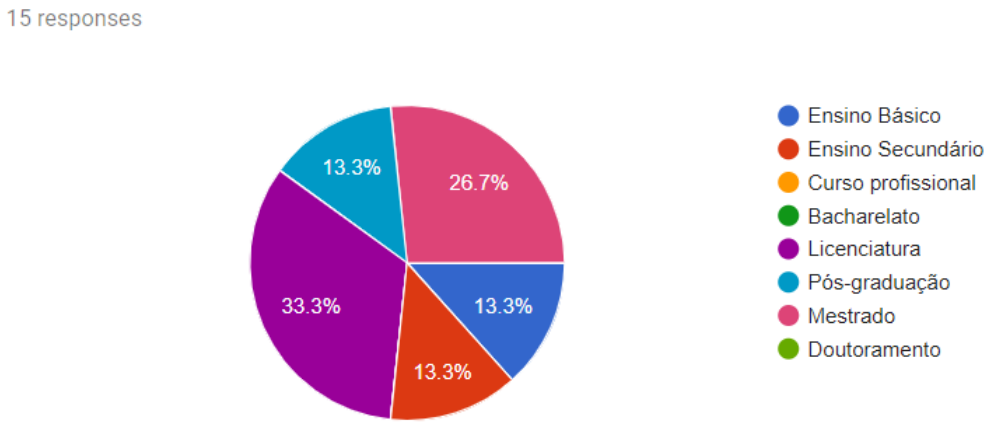
	1	2	3	4	5	
Discordo plenamente	◯	◯	◯	◯	◯	Concordo plenamente

Agradecemos a sua participação!

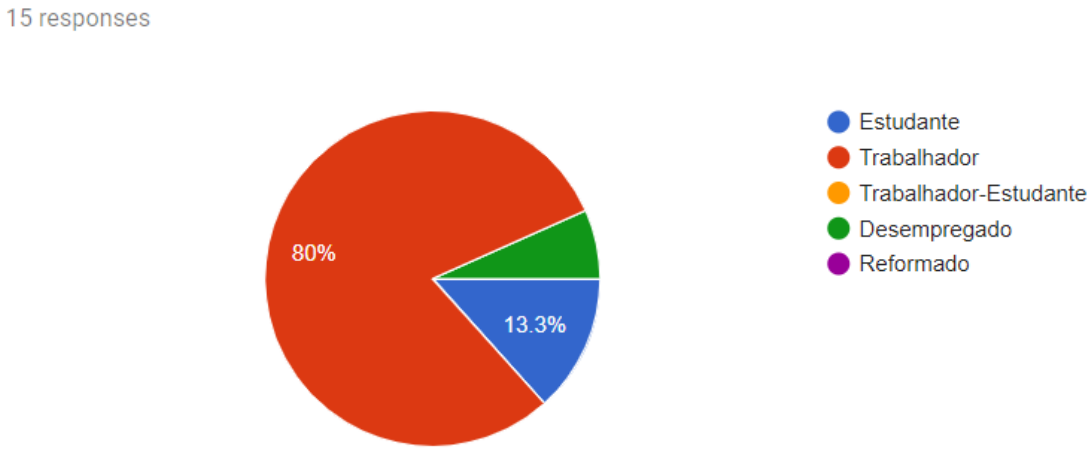
Anexo 3 – Guião da entrevista

Guião da entrevista									
Funcionalidade	Conhece as funcionalidades	Já sabia executar		Lembrou-se de como executar		Erros dados até executar a ação		OBS
		Telecomando	Voz	Telecomando	Voz	Telecomando	Voz	
Ver mais tarde (gravações manuais e automáticas no caso do MEO)								
Agendar um conteúdo								
Seguir um conteúdo (Favorito no caso do MEO)								
Abrir os Meus Conteúdos								
Ver mais tarde								
Agendar um conteúdo								
Seguir um conteúdo								
Abrir os Meus Conteúdos								
Ver mais tarde								
Agendar um conteúdo								
Seguir um conteúdo								
Abrir os Meus Conteúdos								
Ver mais tarde								
Agendar um conteúdo								
Seguir um conteúdo								
Abrir os Meus Conteúdos								
Ver mais tarde								
Agendar um conteúdo								
Seguir um conteúdo								
Abrir os Meus Conteúdos								

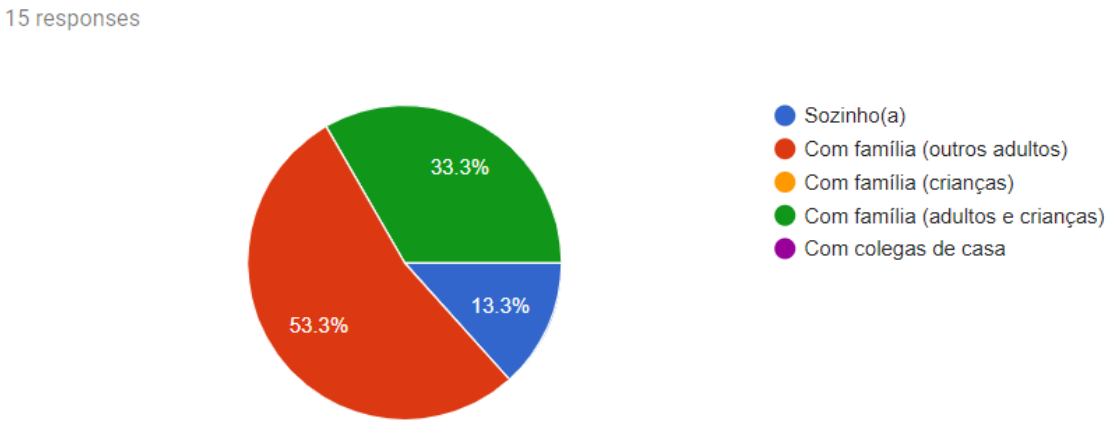
Anexo 4 – Gráfico de habilitações literárias



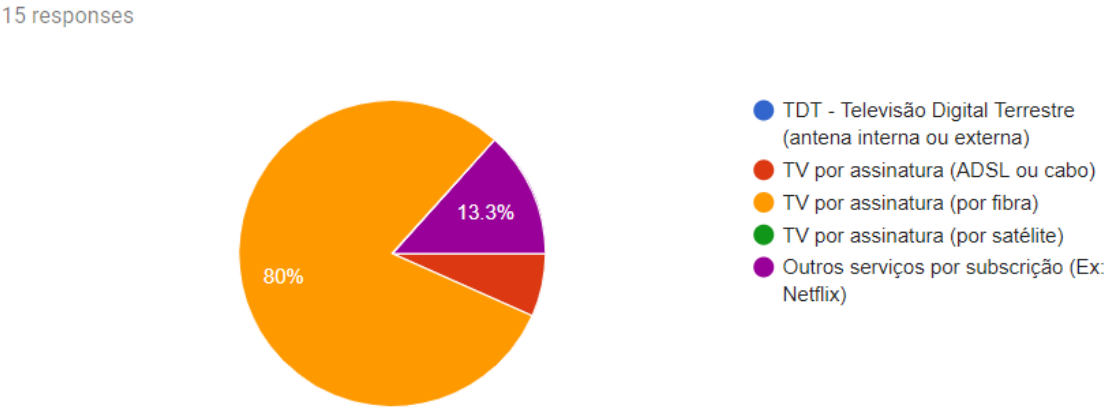
Anexo 5 – Gráfico da situação profissional



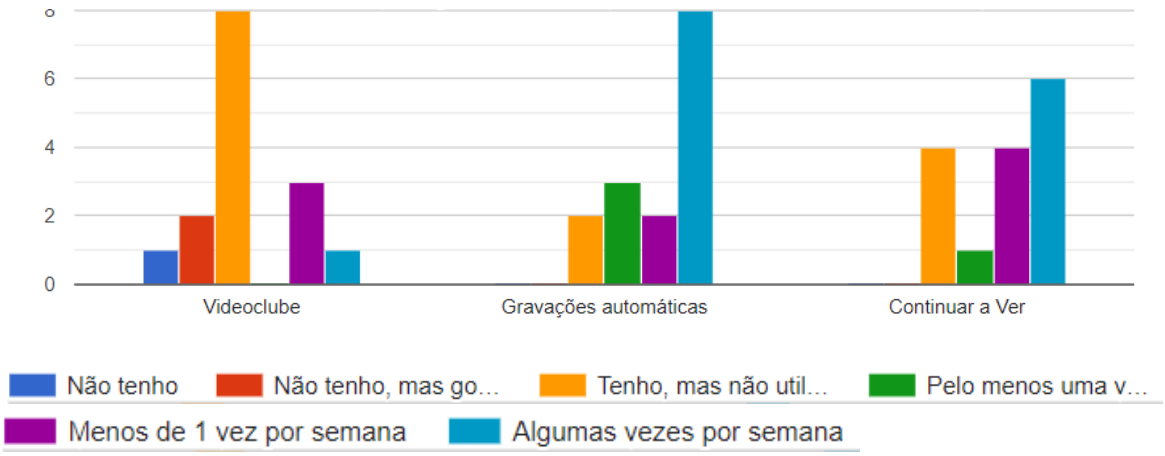
Anexo 6 – Gráfico de visualização de televisão em casa



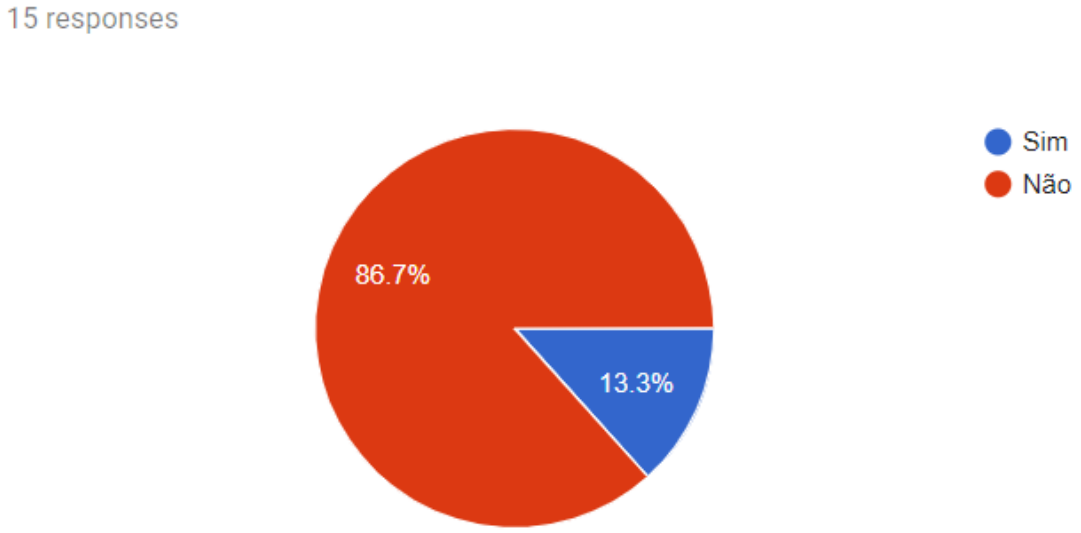
Anexo 7 – Gráfico de tipo de acesso a conteúdos televisivos



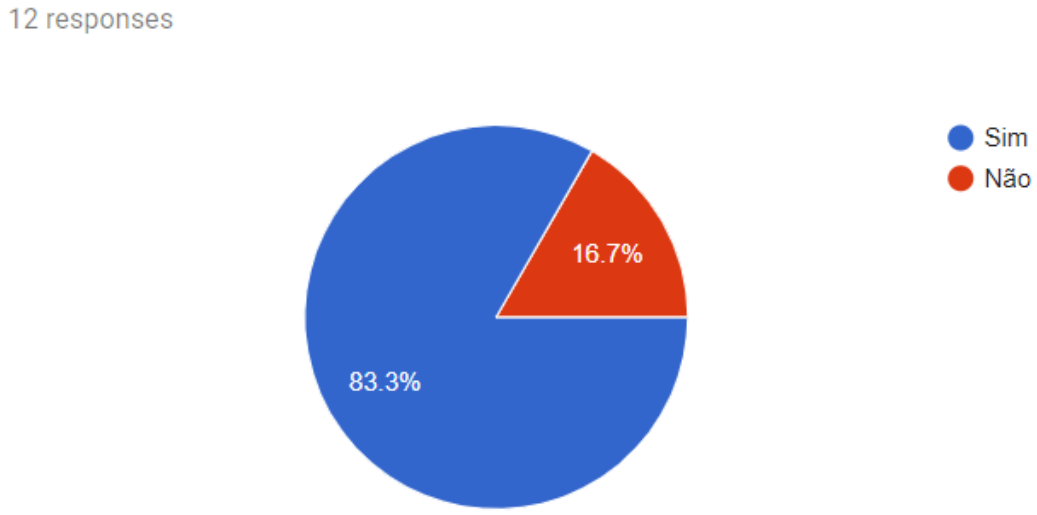
Anexo 8 – Gráfico de frequência de utilização de funcionalidades de televisão



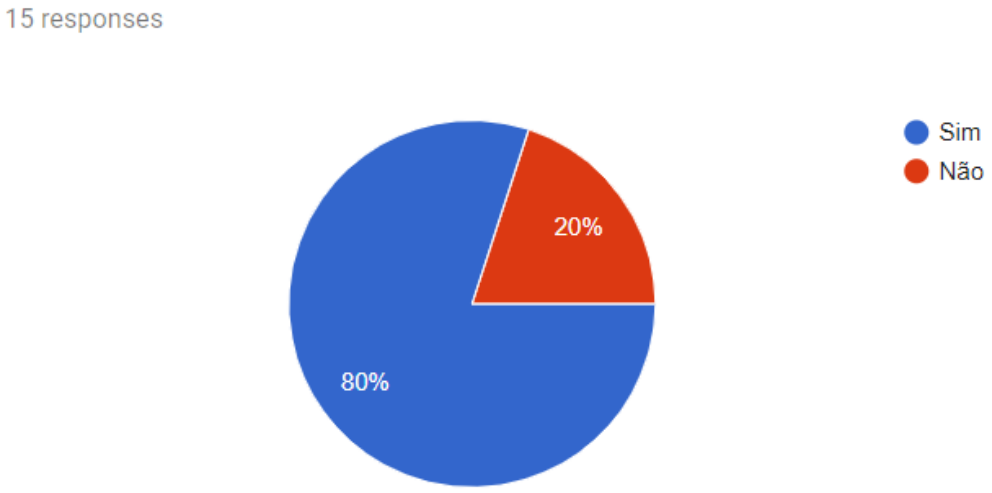
Anexo 9 – Gráfico de utilização de interação por linguagem natural com a televisão



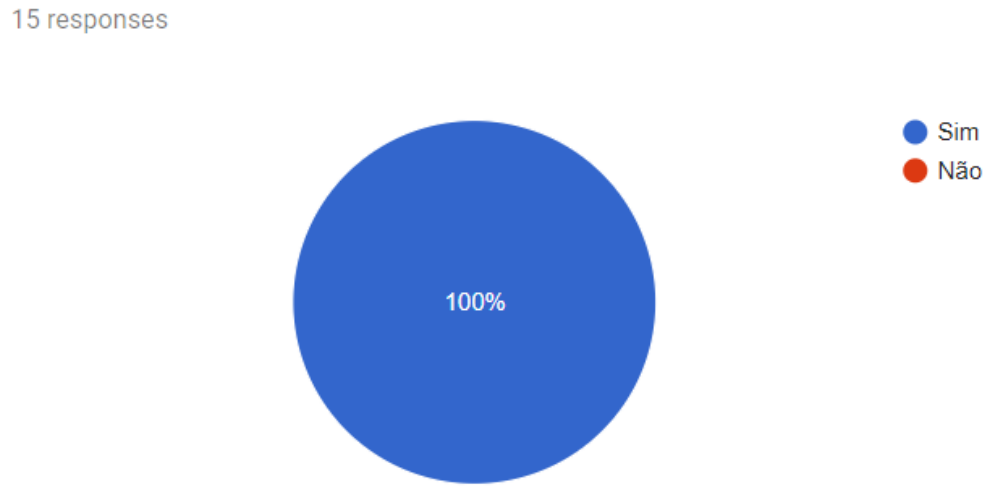
Anexo 10 – Gráfico de interesse em interagir por linguagem natural com a televisão



Anexo 11 – Gráfico de interesse em que o sistema recomende conteúdos de forma proactiva



Anexo 12 – Gráfico de interesse em que o sistema sugira conteúdos que o utilizador segue



Anexo 13 – Gráfico de interesse em que o sistema sugira conteúdos que estão prestes a expirar