EVOLUTION OF RADIO SERVICES IN THE ERA OF VOICE-CONTROLLED DIGITAL ASSISTANTS

P. Casagranda, F. Russo, R. Teraoni Prioletti

Rai, Research & Technology Innovation Centre, Italy

ABSTRACT

The paper evaluates the impact of voice-controlled digital assistants on the evolution of radio services, proposing to apply radio personalization, recommendation and content atomization techniques to the conversational interface. The paper introduces the relevance of smart speakers for radio, and then technically describes a prototype based on voice-controlled digital assistants providing a set of personalized audio services, like keyword-based Audio-On-Demand (AoD) search, semantic similarity search and content-based audio recommendations. A first analysis of the impact of semantically coherent audio segments with metadata, or audio atoms, is also included, leveraging on a subset of the available AoD archive.

The developed prototype service is then tested and assessed with a user evaluation with the purpose to measure the listener's reaction and propensity to access flexible radio services using a voice interfaces.

INTRODUCTION: VOICE-CONTROLLED DIGITAL ASSISTANTS AND RADIO

The voice-controlled digital assistant market is rapidly growing and usage statistics indicate music, news and radio services among the most requested content. On the other hand, an intuitive interface based on voice and natural language radically changes the way listeners access audio content, and indirectly influence the content itself requiring different duration and formats. Currently these digital assistants are generally identified with the physical device, the *smart speaker*, however the core functionality they provide is AI. After this early wave, digital assistants will probably spread widely in a variety of different devices, see the report by Activate (1).

Figure 1 – Smart speakers from Google and Amazon

Recent market analysis estimates show that awareness and ownership of smart speakers is rapidly increasing, with respectively 223 million people being aware of the technology and 65 million people owning one device or more only in the USA in 2019, see Edison

Research and Triton Digital (2) and NPR and Edison Research (3). Currently the biggest players are Amazon and Google, see Figure 1, but from 2014, the year of introduction of the first device from Amazon, several other companies have brought their voice-controlled digital assistants to the market: Apple, Samsung, Huawei, Baidu, Xiaomi, Alibaba and others (all product names and brands are property of their respective owners). The devices are showing the fastest adoption rate of any consumer device, and the functionalities they provide will probably stay with us for a long time, see (1). On the other hand, using smart speakers, listeners often ask for music (90% of first adopters in the last week), news (51%), live radio stations (41%), see (3). Also, a part of smart speaker owners uses them instead of traditional radios (about 40%).

The above scenario makes smart speakers extremely relevant for radio. This fact and the awareness that a third party who owns the digital assistant can act as a gatekeeper, make them both an opportunity and a threat for radio broadcasters. Analysing the potential impact of these devices is therefore useful for radio service evolution. Several European broadcasters have proposed prototypes and services for radio on smart speakers in recent months, and the EBU has created the VOX expert group to foster collaboration.

As a step towards the understanding of how to leverage the power of smart speakers in favour of radio, we designed a prototype, working with content from Radio Rai. The selected platform was Google Home, as it was the first smart speaker launched in Italy in March 2018.

PROTOTYPE SERVICE FUNCTIONALITIES

A prototype service was designed, implemented and assessed to understand the impact of smart speakers on advanced radio services. The service was accessible as an Action, i.e. a custom functionality of the Google digital assistant. The keyword search, semantic similarity search and recommendation engines were borrowed from the personalized linear radio prototype described in Casagranda et al. (4), (5). The functionalities of the prototype are described below.

Live radio and AoD listening. The prototype allowed access to both live and AoD. The last episode of a programme (or the latest news) could be found by mentioning the programme title (title-based search).

Keyword search. The listener could search for AoD from the archive using a keyword. The functionality leveraged the full-text transcripts, provided by the *ASR Engine* of the *Backend Server*, see Section “Prototype Service Architecture”. The ranked list of content returned by the search could be navigated with a “skip” function. The search was especially effective with atomized content, allowing relevant audio segments to be addressed. A keyword search was also used to access specific programme episodes based on their content.

Semantic similarity search. The listener could ask for AoD semantically similar to the current one. Similarity searching allows access to a ranked list of content, based on the comparison of concepts extracted from the transcript of the audio programme. Like keyword search, the semantic similarity search leveraging atomized audio, was more effective, as the concepts included in the audio segment were more coherent, see Gabrilovich and Markovitch (6).

Skipping content. Keyword search allowed easy access a specific audio segment, however, often an entire list of audio segments was returned after a search. The "skip" functionality was selected as a convenient way to go through audio segments with decreasing relevance, avoiding the time-consuming operation of completely listing them for the listener. Skip was implemented to navigate search, similarity and recommendation lists of atomized content.

Content recommendations. The listener could ask for a suggestion of content which would interest him/her. The suggestions were organized as lists of ranked content, created by the content-based *Recommender System* analysing listeners' previous behaviour.

Context-related content. The listener could request context-related AoD (see Ricci et al. (7) for an overview of context-aware recommender systems). Context can be described with a huge number of variables: e.g. time, date, periodicity, location, activity, group. In the prototype we only addressed location, which was automatically retrieved from the smart speaker's API when the local news was requested.

AUDIO CONTENT TYPES

Audio content was prepared to be accessed with smart speakers. The available audio content types are described below.

Live Content. The Action allowed direct access to the live radio streams of the 12 national digital radio services of Radio Rai (from Rai Radio1 to Rai Radio Kids).

AoD. A subset of Radio Rai AoD (e.g. Caterpillar, Wikiradio, 610, Fuorigioco) as well as national and regional news, economy updates and sports news were included in the prototype.

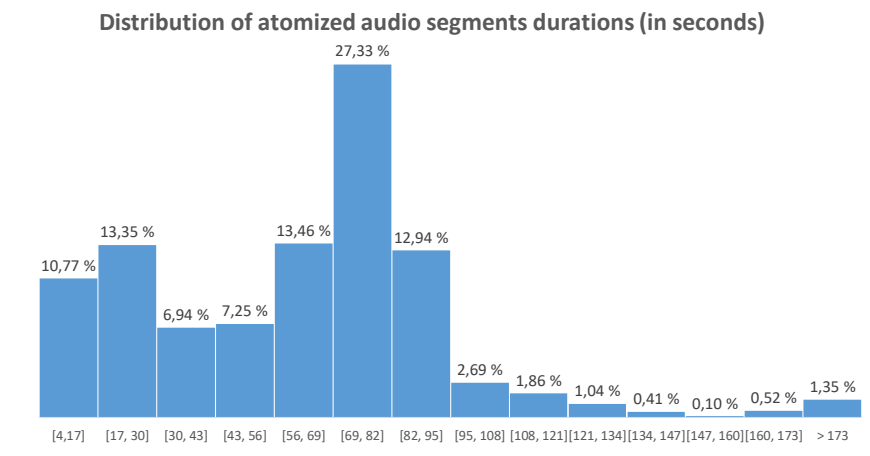

Figure 2 – Atomized audio durations distribution

Atomized Audio. News (general, sports and economy) and a subset of AoD was *atomized*, i.e. segmented in semantically coherent atoms of audio content with rich metadata, including title, EBU Core category according to EBU MIM-AI (8), a short description of the audio, segment information (service, programme, segment number…), and the automatically created audio transcript. The concept was previously applied by the BBC with the Atomized news, see BBC R&D (9), and recently also by the Sveriges Radio in their Play app. Atomization was the only refined manually part of the process, as the automatic cut of the audio was not precise enough to guarantee an acceptable user experience. For the prototype, about one thousand audio atoms were prepared, with the average duration of about 1 minute and duration intervals distribution shown in Figure 2.

PROTOTYPE SERVICE ARCHITECTURE

In order to evaluate the idea of extending radio services to the new dialog based interface, we designed a prototype Action called *Rai CRITS* letting users access Rai Radio services through the Google Assistant. As Radio Rai content is in the Italian language, after some initial tests on Amazon Alexa, we chose the Google platform because it already supported Italian when we started the trial.

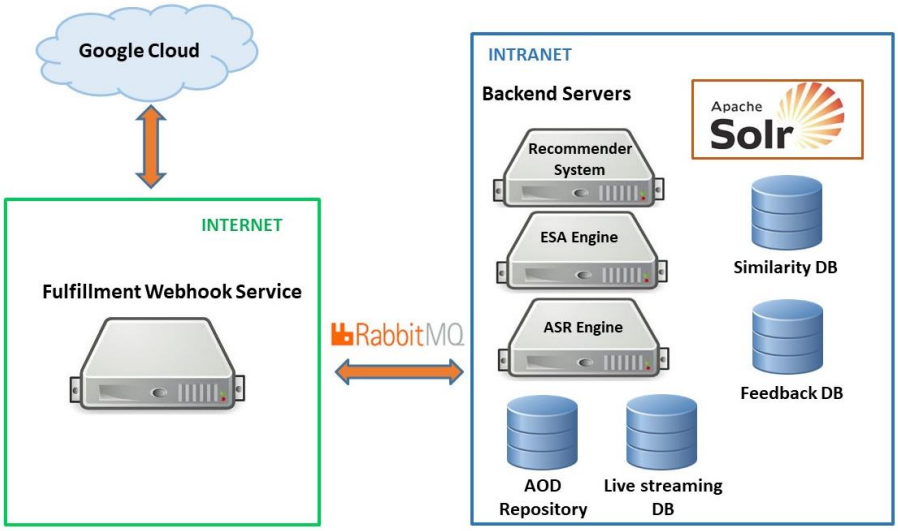

Figure 3 – Service Architecture Scheme

The system architecture can be simplified into two main functional blocks: *User input management*, in Google cloud, and *Request handling and response returning,* in the *Fulfillment Webhook Service* and the *Backend Servers,* see Figure 3.

User input management

A voice-controlled digital assistant allows a task to be performed based on a voice command. Therefore, one of the main components of a smart speaker Action or Skill is the voice interface. We tried to make access to services as simple as possible and the dialog between users and the digital assistant, as intuitive as possible.

In this first prototype version we built a very simple conversation interface. Improving the quality of the voice interaction would require a specific, substantial effort.

To develop *Rai CRITS Action* we used Google's *Action on Google,* the platform used to create the project, including the *Dialogflow* tool for defining conversational interfaces and performing the natural language processing. *Dialogflow* allows the underlying goal or task required by a user, to be defined. Called *Intent,* examples are: listening to music or turning the lights on. Each different service that the *Rai CRITS Action* provides is handled by a different Intent: whenever the user asks a question, the corresponding Intent is triggered.

Moreover, when you build an Action for the Assistant, you can design your conversations supporting for example audio or audio/visual devices. Visual devices can be useful to present detailed information and easily access lists of items. For our test, we used only the screen-less Google Home and Google Home Mini, excluding other devices.

Request handling and response returning

Whenever a user sends a request to Google Assistant, the corresponding Intent, on the *Fulfillment Webhook Service*, is triggered to process the answer. Every Action has a corresponding *fulfillment,* a component defining the logic for handling intents and

dynamically constructing responses to send back to Google Assistant. Our fulfillment service is hosted on Rai web servers, see Figure 3.

The Fulfillment Webhook Service securely receives and handles Intent requests and builds a response based on data provided by the *Backend Servers*. Each response contains all the information to play audio (AoD and live streaming). If the response contains more than one item, the skip function is also managed by the *Fulfillment Webhook Service*. The response generated by this server is then sent to Google servers.

The Backend Servers contain the logic to select the audio content which satisfies user requests and include all the information about audio content and the user's history and behaviour. The main databases deployed on the *Backend Servers* are the following:

- AoD Repository, which contains all AoD metadata and references to the audio segments.
- Live streaming DB, which contains all the references to Radio Rai live streams.
- Feedback DB, where the implicit user's feedbacks are stored.
- Similarity DB, where similarities between one AoD and all the others are stored in a matrix.

AoD metadata, user history and all available context information are taken into account by the *Backend Servers* to build the response. Our prototype shares most of the logic with the already-implemented *hybrid content radio framework* app (4). In this way, the same logic is used to handle requests from different platforms and interfaces.

The *Backend Servers* include an automatic speech recognition server (**ASR Engine** in Figure 3) automatically analyzing all new AoD coming from Radio Rai, storing the transcripts and indexing them using an **Apache Solr** server. The analysis is performed both on full-length AoD and on atomized content.

The **ESA Engine** is used to perform Explicit Semantic Analysis (ESA) on the transcripts of AoD and audio atoms. ESA allows representation of an unrestricted natural language text as a concept vector, that is a vector in a *high dimensional space of concepts* from a knowledge base, see (5), (6). Specifically, the vector space base used for the representation has been taken from Wikipedia. The concept vector representation is used for semantic similarity comparisons between different AoD. As similarity analysis computations are time-consuming and infeasible for real-time services, AoD similarity scores are calculated as soon as a AoD is available and the resulting matrix is stored on the **Similarity DB**, making the calculation time independent of the number of user requests.

The backend also includes a **Recommender System**, performing content-based recommendations leveraging the semantic similarity engine. Using the *ASR Engine* and *ESA Engine*, each AoD is represented as a concept vector. On the other hand, the recommender system needs to know a detailed user profile in order to provide useful recommendations, and for this purpose implicit user feedback is considered. Whenever a user listens to an AoD that comes from a *Keyword search* or a *Semantic similarity search*, a positive implicit feedback related that AoD is stored. In this way, each *i-th* user is represented by $\boldsymbol{ESA(user_i)}$, a combination of ESA concept vector of the AoD searched and

listened. Finally, for each user, a score is assigned to each AoD in the audio content repository based on the similarity to $ESA(user_i)$, and the score assignment is used to create an ordered list of candidate AoD. Recommended clips for a user are then taken from the ordered list, choosing the candidates with highest scores. The list of recommended AoD is included in the response.

All prototype functionalities described in the previous Section are implemented in the *Backend Servers*. We describe them in detail in the following paragraphs.

Live radio and AoD listening. The *Backend Servers* perform a title-based AoD search to find the most recent news or AoD episodes. If a live radio service is requested, the servers return a reference to the proper stream. If an atomized version of the required AoD is available, the response lists all the audio atoms belonging to the requested AoD.

Keyword search. Given a keyword, this functionality performs a full-text search of that keyword across the whole audio content archive, based on the metadata and the transcript extracted by the *ASR Engine* and indexed by *Apache Solr*.

As a result, the service returns the most relevant list of items. In order to have an even better result, it's also possible to add manual metadata to every AoD.

Semantic similarity search. This functionality returns the list of AoD similar to a specified AoD. Similarity values are retrieved from the similarity matrix stored in the *Similarity DB*.

Context-related content. Location was automatically retrieved from the smart speaker's API and the latest local news is returned. If an atomized version of the required news is available, the response lists all the audio atoms belonging to the retrieved news.

Content Recommendations. This functionality returns the list of recommended AoDs for a specific user. Recommendations can usually take advantage of several algorithm classes: collaborative filtering, content-based filtering, knowledge based filtering, social based and others, see (7). During the trial we only took advantage of the content-based *Recommender System* previously described.

USER EVALUATION

The user evaluation has been carried out with three groups, two test groups and one control group, a total 186 people, all of them diverse in terms of gender. *Group A*, the first test group, included 5 classes of high school students and professors, a total of about 105 people coming from a number of schools in Italy interviewed between March and June 2019. They attended a guided demonstration, which gave them the opportunity of asking questions of the development team and interacting with the prototype. They finally compiled a survey about their experience.

Group B, the second test group, included 51 people from different Rai business units, including administration, advertisements management, ICT, research and human resources. They were briefly introduced to the general smart speaker functions with simple examples, and answered a preliminary survey. Then, the interviewer briefly explained the features of the voice-controlled prototype, describing the set of functionalities and how to activate them. Finally, the volunteers tested the prototype, making a number of requests for each functionality, for about 30 to 45 minutes. All feedback about the prototype and the

smart speaker was collected in a survey about the experience. For this group, the logs of the interactions they made with the device were also recorded and subsequently analysed.

The feedback from both groups was analysed, and further indications were extracted from the logs of the interactions.

Finally, *Group C*, a third group of 30 people from different Rai business units, was used as a control group to check if results of the other two groups were statistically significant. Group C volunteers only tested already-available Rai live radio streams, *without having access to the prototype's functionalities*.

KEY FINDINGS AND CRITICALITIES

In the following discussion we will refer to the *Group B* results, if not otherwise specified. Before using the prototype, the respondents reported that they were mainly interested in requesting music (25%), weather forecasts (24%), latest news (18%). Live radio listening accounted for about 10%.

An indication of the smart speaker impact is evidenced from t answers to the question "How do you rate smart speakers when listening to radio content?", before and after having played with the prototype, see Figure 4. The number of respondents who judged the smart speaker as useful almost doubled after the test. Statistical significance was evaluated using Pearson's chi-square test, which gave a value of P=0.006, i.e. the

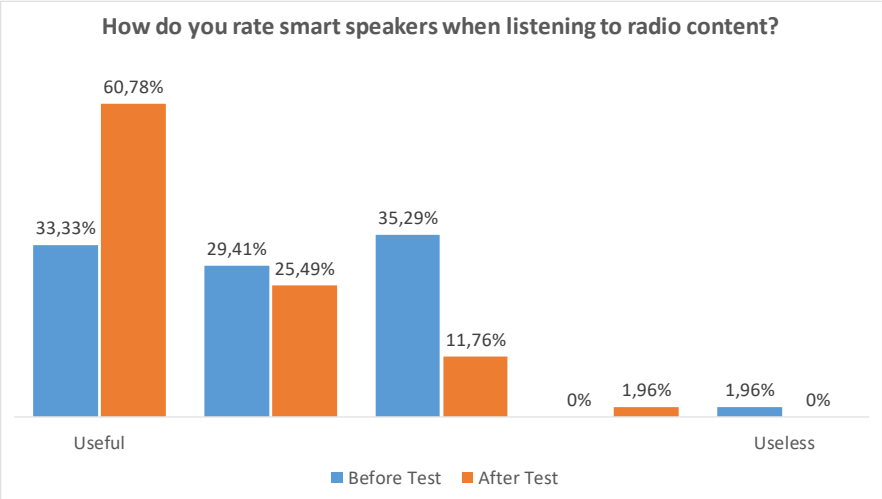

Figure 4 – Evaluation of smart speakers for radio content

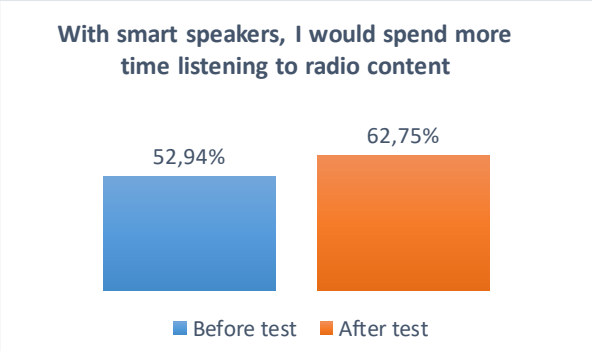

Figure 5 – Listening time

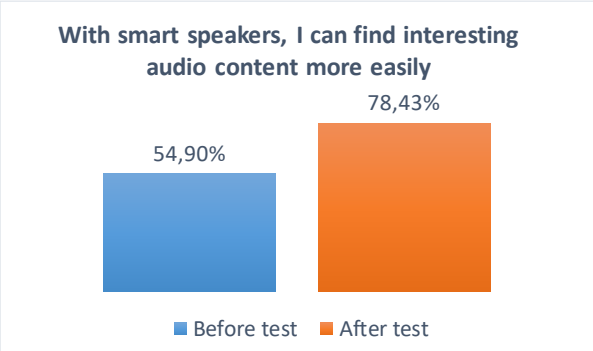

Figure 6 – Content accessibility

difference is statistically significant. This indication was clear among the volunteers and similar Groups A and B.

After the test, about 63% respondents said they would spend more time listening to radio content thanks to the smart speakers, see Figure 5. In this case the difference between the answers before and after the test is statistically less significant (chi-square P=0.3), however a relevant outcome is that more than half of respondents think smart speakers would increase the time spent listening to radio (98% of respondents would not decrease the time spent listening to radio). Another indication came from the question "With smart speakers, I can find interesting audio content more easily", agreed by 78% of the respondents: they were 55% before the test, a neat increase unlikely due to chance (chi-square P=0.01), see Figure 6.

On the other hand, *Group C (the control group) results did not show any statistical significance for questions of Figures 4 to 6.* Listening Rai's live radio streams with a smart speaker did not significantly change the perception of users towards radio, confirming a strong difference compared to the results of Group B, which evaluated the prototype.

Moreover, in both *Group A* and *Group B* about 80% of the respondents agreed that requesting a specific podcast based on its title (title search) was useful with smart

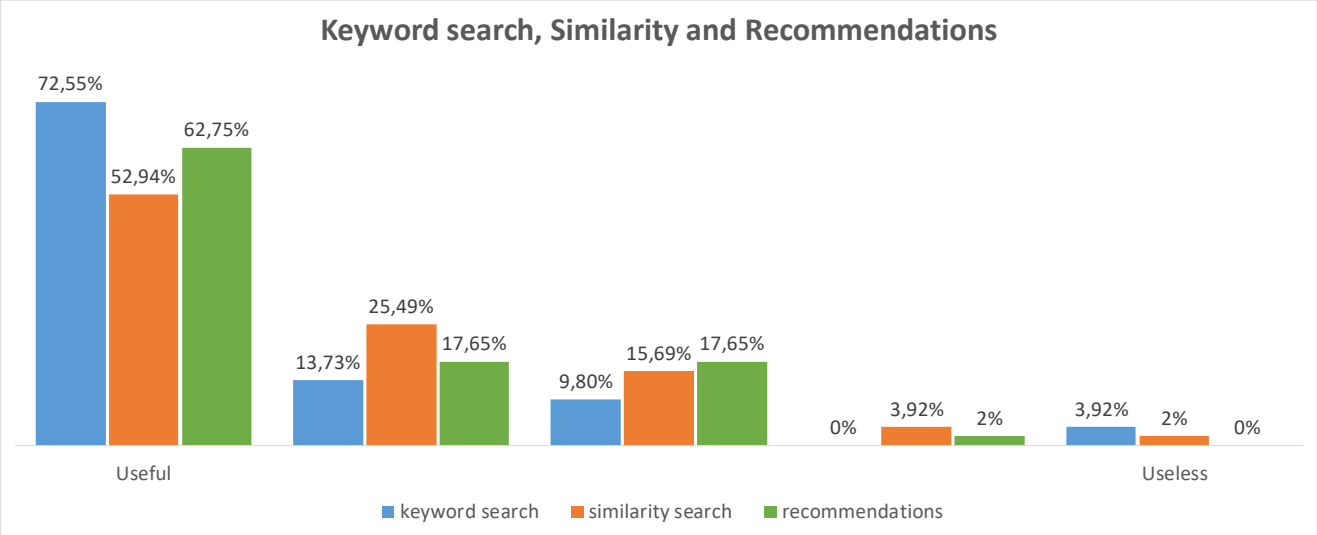

Figure 7 – Evaluation of keyword search, similarity and recommendations

speakers.

Keyword search, *similarity search* and lists of suggested AoD (*recommendations*) were generally evaluated as very useful, see Figure 7. The keyword search functionality was the most appreciated (about 73%), and in the discussion after the survey several respondents confirmed the search was a surprising feature. Also lists of suggested AoDs and similarity

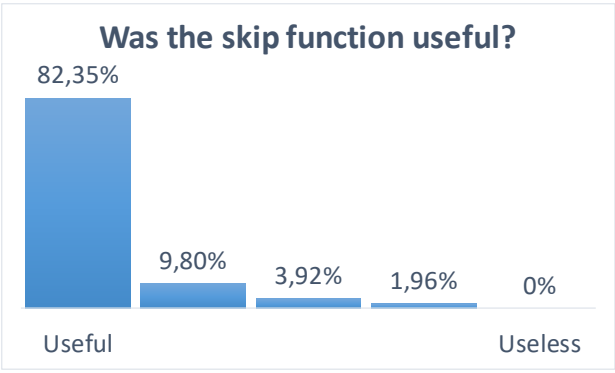

Figure 8 – Evaluation of the skip function

searches were appreciated (with about 63% and 53%). Some respondents pointed out that suggestions could be refined by including e.g. the listener's mood, asking "something that's makes me feel better", or "something quiet or energizing".

The short segments of atomized audio were appreciated (about 69%), and the skip function to navigate atomized content was judged useful (about 82%), and proved to be an effective compromise to quickly reach interesting content after a search, see Figure 8.

An interesting indication came from the question about what kind of content they would listen with a smart speaker, semantically polarizing it between live radio and AoD. After the test, respondents showed a preference for podcast content over live radio content (31% vs 22%), and the difference was even more clear in *Group A* (45% vs 23%), as shown in Figure 9. The overall difference between the two groups is not statistically significant (chi-square P=0.47). *So podcasts were preferred on live radio by both groups*. This seems to be related to the possibility of quickly searching for interesting content. The duration of each atomized news segment (see Figure 2) was considered adequate by the majority of respondents (61%). Context-based audio was also rated as useful (about 63%), although it was only tested with Rai regional news.

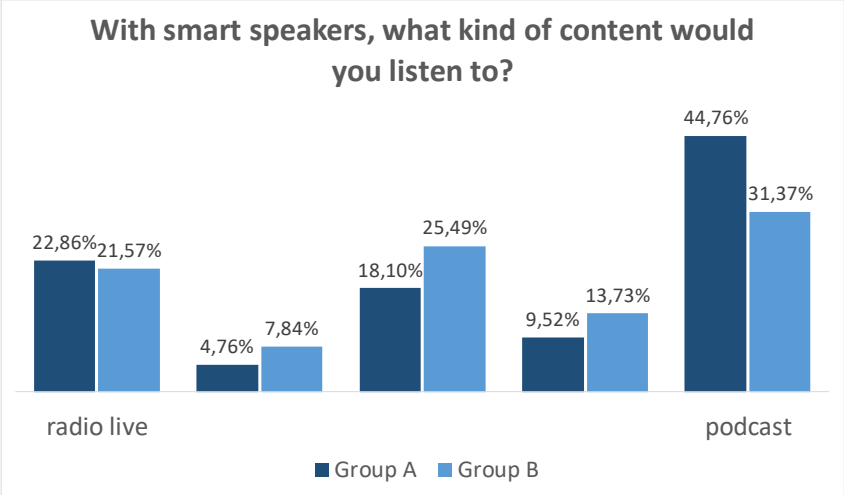

Figure 9 – General evaluation of smart speakers for radio

The prototype evaluation also allowed the gathering of a number of *criticalities* that we report here. A major drawback was the rigidity of the requests scheme. Activating the Action was unintuitive, requiring the "speak with RAI CRITS" command. Sometimes changing a word in the request could lead to an error. The listener had to be trained to access the advanced functionalities (search, similarity, recommendations). Refining the interaction with the user requires great attention to the conversational interface design with a careful analysis of user requests, as well as a more flexible interpretation of natural language.

Other drawbacks were related to the audio content selection: some respondents suggested adding the possibility of selecting episodes by specifying a date or time (e.g. "*let me listen to yesterday's Wikiradio*"). In general, the visual help of a display in the interface could make search and selection faster and more powerful.

Some respondents pointed out substantial concerns about privacy: the smart speaker can record requests and very personal data, even more than a search engine, and a strict regulation and control of the usage of the collected data is critical, see Day et al. (10).

CONCLUSION

This paper has analysed the impact of voice-controlled digital assistants on radio services focusing on flexible and personalized radio services and on atomized content, leveraging a specifically-built prototype service and a subset of AoD from the Radio Rai audio archive. The prototype was evaluated by two panels of volunteers including more than 150 people, pointing out some strengths and criticalities of this technology. Full-text keyword search in audio podcasts was the most appreciated advanced functionality by both groups, followed by recommendations and similarity search. An interest towards AoD and atomized content emerged, although their actual impact on the listening time requires further investigation. In general, the availability of semantically highly-coherent short segments of audio was appreciated, so producing atomized content seems to be important for voice-controlled radio services.

Criticalities included very general limitations due to the platform adopted, privacy concerns and very specific feedback related to the limited availability of audio content.

The final feedback from the respondents, both students and employees, independently of their previous knowledge of the technology, was generally very positive, giving an encouraging indication about the usage of advanced features for radio on smart speakers.

BIBLIOGRAPHY

[1] Activate, 2017. Activate Tech & Media Outlook 2018, https://www.slideshare.net/ActivateInc/activate-tech-media-outlook-2018

[2] Edison Research and Triton Digital, 2019.The Infinite Dial. https://www.edisonresearch.com/infinite-dial-2019/

[3] NPR and Edison Research, 2018. The Smart Audio Report. https://www.nationalpublicmedia.com/smart-audio-report/latest-report/

[4] P. Casagranda, M. L. Sapino e K. S. Candan, 2017. Context Aware Proactive Personalization of Linear Audio Content, In Proceedings of the 20th International Conference on Extending Database Technology (EDBT), 574-577

[5] P. Casagranda, F. Russo, R. Teraoni Prioletti, 2018. Personalizing Linear Radio: Model and User Evaluation, In Proceedings of the IBC 2018

[6] E. Gabrilovich, S. Markovitch. 2007. Computing semantic relatedness using wikipediabased explicit semantic analysis. In Proceedings of the International Joint Conference on Artificial Intelligence, 7, 1606-1611

[7] F. Ricci, L. Rokach, B. Shapira, 2015. Recommender Systems Handbook. Springer

[8] EBU MIM-AI, 2019. EBU Core Metadata Set (EBUCore). In EBU Tech 3293 v. 1.9

[9] BBC R&D, Atomised news - with BBC R&D http://bbcnewslabs.co.uk/projects/atomised-news/

[10] M. Day, G. Turner, N. Drozdiak, 2019. “Amazon Workers Are Listening to What You Tell Alexa”. https://www.bloomberg.com/news/articles/2019-04-10/is-anyone-listening-to-you-on-alexa-a-global-team-reviews-audio