Building Intelligent Digitial Assistants for Speakers of a Lesser-Resourced Language

Jones, Dewi; Cooper, Sarah

Proceedings of the LREC 2016 Workshop “CCURL 2016 – Towards an Alliance for Digital Language Diversity”

Published: 23/05/2016

Publisher's PDF, also known as Version of record

LREC 2016 Workshop

CCURL 2016

Collaboration and Computing for Under-Resourced Languages: Towards an Alliance for Digital Language Diversity

23 May 2016

PROCEEDINGS

Editors

Claudia Soria, Laurette Pretorius, Thierry Declerck, Joseph Mariani, Kevin Scannell, Eveline Wandl-Vogt

Building Intelligent Digital Assistants for Speakers of a Lesser-Resourced Language

Dewi Bryn Jones[1], Sarah Cooper[2]

[1]Language Technologies Unit, [2]School of Linguistics and English Language
Bangor University, Bangor, Wales, UK
E-mail: {d.b.jones, s.cooper}@bangor.ac.uk

Abstract

This paper reports on the work to develop intelligent digital assistants for speakers of a lesser-resourced language, namely Welsh. Such assistants provided by commercial vendors such as Apple (Siri), Amazon (Alexa), Microsoft (Cortana) and Google (Google Now) are allowing users increasingly to speak in natural English with their devices and computers in order to complete tasks, obtain assistance and request information. We demonstrate how these systems' architectures do not provide the means for external developers to build intelligent speech interfaces for additional languages, which, in the case of less resourced languages, are likely to remain unsupported. Consequently we document how such an obstacle has been tackled with open alternatives. The paper highlights how previous work on Welsh language speech recognition were improved, utilized and integrated into an existing open source intelligent digital assistant software project. The paper discusses how this work hopes to stimulate further developments and include Welsh and other lesser-resourced languages in as many developments of intelligent digital assistants as possible.

Keywords: intelligent digital assistants, speech recognition, lesser resourced languages, Welsh

1 Introduction

It is increasingly possible, as a consequence of recent advancements in speech recognition, machine translation and natural language processing and understanding, for users to engage with their devices and computers. They do this via intelligent speech interfaces in order to command and control as well as to receive answers to questions voiced in natural language.

There are four main commercial platforms driving this change, namely Apple Siri, Google Now, Microsoft Cortana and Amazon Alexa. To date these provide their powerful capabilities in English and to a lesser extent some other major languages. There is little evidence so far that they are likely to extend their choice of languages to the long tail of smaller languages, including Welsh, in the near future. Furthermore there are no means for external developers to adapt these systems for any new language.

Indeed languages with smaller numbers of speakers often find themselves lagging in digital innovation including language technologies. As such they are lesser-resourced with regard to the availability and interest in funding. However the Welsh Government through its Welsh Language Technology and Digital Media Fund have since 2012 have followed a strategy to develop 'more tools and resources… to facilitate the use of Welsh, including in the digital environment" (Welsh Government, 2012; 45) as well as develop "new Welsh language software applications and digital services" (Welsh Government, 2013; 12).

With funding from the Welsh Government as well as S4C, (the Welsh-language public service television channel), we have established a project called 'Seilwaith Cyfathrebu Cymraeg' (*Welsh Language Communications Infrastructure*) (Jones and Ghazali, 2016). This project aims to ensure that Welsh language users are not excluded from continued developments in human computer interaction by improving current Welsh speech recognition and applying its capabilities in a prototype Welsh language intelligent digital assistant application. The project is limited to 8 months in duration due to the initial funding programme.

In addition the project will make all resources and software available under very permissive open-source licenses via the Welsh National Language Technologies Portal infrastructure (Prys and Jones, 2015). This provides unrestricted usage to developers involved in commercial, education and volunteer activities within the lesser resourced language community. It also provides unrestricted usage for global companies who wish to extend their range of languages supported in any multilingual intelligent digital assistants.

2 Approach

The 'Seilwaith Cyfathrebu Cymraeg' project initially evaluated the four main commercial intelligent digital assistant platforms that are responsible for popularising a new mode of human computer interaction (Apple Siri, Amazon Alexa, Microsoft Cortana and Google Now) Each platform is complemented by APIs (Application Programming Interfaces) and SDKs (Software Development Kits) that each company is eager for developers to utilise in their commercial products and services. This enables each platform to have its capabilities extended and presence widened into third party apps and products. We initially investigated whether these systems would provide an opportunity to extend the range of supported language (Ghazali et al., 2015)

A generic software architecture and flow of events was realised during the investigation as seen in Figure 1.

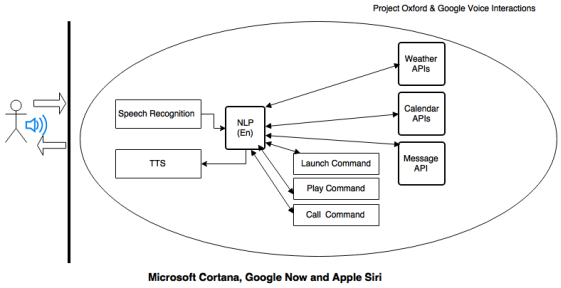

Figure 1 - Generic Architecture of an Intelligent Digital Assistant

A user speaks in natural language to the device or computer:

1. A speech recognition engine captures the audio and converts the user's spoken wish into text. The text is handed over to a natural language processing (NLP) component. For example the user asks: 'DO I HAVE ANY MEETINGS TODAY?'
2. The NLP component understands from the text what the user intends or wants and thus identifies which component, module or API is able to fulfil the request. For example, the NLP component may recognize a question from 'DO I HAVE' and recognizes 'MEETINGS' and 'TODAY' as keywords associated with time and calendar.
3. The NLP component communicates with the identified module or API according to its interface specification. For example it constructs and sends a message to the Calendar API:

   ```
   getEvents(type=meeting,date=today)
   ```

4. In cases where an answer is required, the NLP accepts a response from the obliging module or API and generates a sentence with results included. For example it may receive a result in a format such as JSON:

   ```
   {"success": true, "events": [
   {"date": "2015-09-25",
   "time":"10:30am", "name":"Meeting to
   discuss a new Welsh language
   project","location":"Bangor
   University, Ogwen Building, Room
   234"},{"date": "2015-09-25",
   "time":"12:30pm", "name":"Lunch with
   Delyth","location":"Terrace
   Restaurant, Bangor University"} ]}
   ```

   From which the NLP constructs the sentence:

   Yes you do. At 10:30 this morning you have a meeting to discuss a new Welsh language project in Bangor University, Ogwen Building, Room 234. Then at 12:30 you have lunch with Delyth in the Terrace Restaurant, Bangor University.

5. The natural language sentence result is handed to a text to speech engine for voicing back to the user. For example:

   "*Yes you do. At half past ten this morning you have a meeting to discuss a new Welsh language project in Bangor University, Ogwen Building, Room two hundred and thirty four. Then at half past twelve you have lunch with Delyth in the Terrace Restaurant, Bangor University.*"

We became aware that these commercial architectures have their speech recognition and natural processing components encapsulated into one super-component. As a consequence they only provided access in the language that the speech recognition component supports. This linguistic limitation may be necessary for a functional consistency but does not allow for external developers to integrate support for additional languages with the aid of alternative language technologies that could still leverage the capabilities of the commercial offerings.
A number of other intelligent personal assistant platforms exist. Their suitability as a basis for building a Welsh language digital assistant is feasible only if their architectures are more granular and open and which can fulfil the following criteria:

- A Welsh language speech recognition engine can be integrated
- The NLP for understanding texts from voiced requests can be either adapted or replaced
- Responses can be provided via Welsh language text to speech
- APIs and modules that implement capabilities and fullfill tasks but which are based in English language usage can still be included with novel integration of Welsh to English in requests and English to Welsh machine translation in responses

Figure 2 illustrates a desirable architecture.

The best open alternative candidate was found to be Jasper (Marsh and Saha, 2014). Jasper is a very simple Python application which already integrates a number of speech recognition and text-to-speech engines and provides an easy mechanism for developers to extend its capabilities via simple addition of modules written as Python scripts. Jasper is able to run completely locally on small computers such as Raspberry Pis.

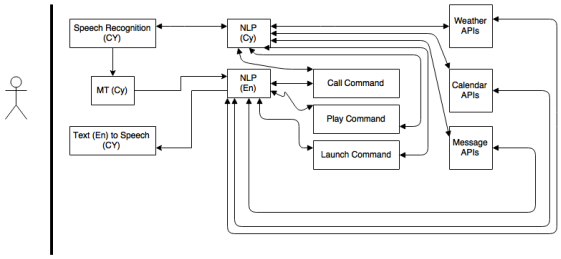

Figure 2 - Desirable Architecture for Building a Welsh Language Intelligent Personal Assistant

Also worthy of consideration, due to the availability of source code and/or sufficient modularity and openness of APIs, were SIRIUS (Hauswald et al., 2015) and Wit.ai (wit.ai, n.d). They contain more complex language technologies and are considerably more complex.

These open alternatives are able to support an incremental development strategy. Initially Jasper can be used for developing a simple speech recognition system in conjunction with an elementary intent evaluator supporting a limited number of domains in Welsh (such as asking the time, news or weather). Subsequently more sophisticated and intelligent capabilities can be developed through further iterations.

Such an approach is compatible to the work to improve the Welsh language speech recognition component and extend its vocabulary and grammar complexity in order to support a growing number of possible requests and questions.

3 Speech Recognition Improvement

Prior to the 'Seilwaith Cyfathrebu Cymraeg' project, a significant amount of work on Welsh language speech recognition had been done in the GALLU (Gwaith Adnabod Lleferydd Uwch, translation: Further Speech Recognition Work) project which built on earlier work on basic speech recognition project during 2008-9 (Cooper et al., 2014). Some earlier work on speech technology work had also been conducted under the WISPR (Welsh Irish Speech Processing Resources) project (Prys et. al., 2004).

The GALLU project succeeded in developing new letter to sound rules, a collection of prompts covering all Welsh language phonemes and subsequently a speech corpus (Cooper et. al., 2015) collected via a crowd sourcing iOS and Android app called Paldaruo (Jones, 2015). Elements of the corpus were successfully used to train HTK acoustic models and their inclusion in an open source speech recognition decoder engine (Julius) that could control the movement of a toy robotic arm (Aonsquared, n.d). The speech recognition component of the 'Seilwaith Cyfathrebu Cymraeg' project would be a continuation of this previous work and means that Julius and HTK would continue to be the basis for developing Welsh language speech recognition.

Improving speech recognition for Welsh would mainly consist of training new acoustic models for all Welsh language phones from the entire Paldaruo speech corpus. The corpus had grown in size since 2014 to contain contributions from 410 speakers. This was in contrast to training during the GALLU project where data from only 20 speakers was used along with a subset of phones. We improved the training scripts from the GALLU project and packaged them with the HTK into a user friendly and portable environment for speech recognition development using Docker. Docker allows for acoustic model training to be easily shared and consistently reproduced amongst other researchers and developers. Scripts were added for downloading the Paldaruo Speech Corpus, prompts text, speaker metadata (e.g. age and accent) as well as a Welsh pronunciation lexicon. We also added scripts that package the acoustic models for use in decoders such as Julius.

At the start of the GALLU project we had not anticipated the automated testing of acoustic models and as such had not collected extra recordings from each speaker for a test corpus. It was left to this project and the Docker based HTK training environment to follow a widely considered 'bad practice' of using training data as test data. A word loop grammar was used so that a test could expect any word after a given word. However this approach was beneficial in evaluating and improving the acoustic models. An initial reproduction and testing of the robotic arm control application from GALLU using the new Docker HTK and scripts environment (Iteration 0, Figure 3,4) showed that the approach was useful in validating contributions from speakers.

Iteration	Speakers	Description
0	20	All contributions used in training GALLU robotic arm control
1	410	All contributions
2	177	All contributions noted as Central and South Wales accents.
3	210	All contributions noted as North Wales accents
4	1	Recording from one contributor validated and verified by ear
5	35	All complete contributions (160 speakers) tested individually to have a Word Accuracy above 70%
6	88	All contributions (410 speakers) tested individually and seen to have word accuracy above 70%
7	88	As 6 but with an improved clustered triphone question for Welsh (tree1.hed)

Figure 3 - Acoustic Model Training Iterations

Iteration	Word Accuracy	Sentence Accuracy
0	90%	42%
1	20%	0%
2	26%	0%
3	24%	0%
4	75%	7%
5	92%	24%
6	91%	19%
7	92%	20%

Figure 4 - HTK Results on accuracy for each iteration

Thus we began by training new acoustic models with all recordings from all speakers (Iteration 1, Figure 3,4). Initially scores were disappointing (Word Accuracy 20%). While attempts at distinguishing between accents did improve word accuracy scores to some extent (Iteration 2, 3 Figure 3,4), we did not deem it sufficient enough to base further iterations on a partitioned speech corpus.

It could be argued that this was to be expected given that the speech corpus was crowd sourced and we had little control on the quality of contributions. Furthermore there could be no human involvement in quality assurance and verification of recordings due to the resources available (given that audio files numbered into their thousands).

However an experiment to train and test acoustic models based on one speaker's contribution, along with amendments to the training and testing scripts, provided a breakthrough which, according to our testing strategy, provided much improved word and sentence accuracy scores (Iteration 4, Figure 3, 4). Subsequently we trained and tested every contribution individually to obtain word accuracy scores for each speaker. This resulted in a wide variety of scores across speakers. All contributions found to have word accuracy scores above 70% were considered better quality.

Of the 410 individuals who had used the Paldaruo app, only 136 had recorded all 43 prompts. We assessed these contributions on an individual basis and 35 of these were found to have a word accuracy score above 70%. When combined together to create speaker independent acoustic models, the word and sentence accuracy improved to over 90% (Iteration 5 Figure 3, 4).

When considering all contributions, regardless of the number of prompts recorded, we found 88 speakers had word accuracy scores higher than 70%. When combined together to create speaker independent acoustic models there was a slight decrease in comparison to the previous iteration in both word and accuracy scores (Iteration 6, Figure 3, 4).

We improved our HTK decision tree clustering script file for Welsh (the language specific tree.hed file) which groups all Welsh phonemes according to their acoustic classes. Our results consequently improved word and sentence accuracy results by 1% (Iteration 7, Figure 3, 4).

4 An Early Prototype Welsh-language Intelligent Digital Assistant

With acoustic models deemed sufficient in quality, the project was able to implement its first iteration of a Welsh language intelligent digital assitant which would support answering questions and fulfill tasks in the domains of news, weather, time, music, proverbs and jokes.

We developed simple grammar and vocabulary files for Julius in order to produce the first release of a Welsh language speech recognition component - julius-cy (Jones and Cooper, 2016). It aims to recognise all possible means of requesting information (such as news, weather and time) in complete and natural sentences. For example:

BETH YDY'R TYWYDD HEDDIW?
What's today's weather?

BETH YW TYWYDD YFORY?
What's tomorrow's weather?

BETH YW'R NEWYDDION?
What's the news?

FAINT O'R GLOCH YDY HI?
What time is it?

CHWARAEA GERDDORIAETH CYMRAEG
Play Welsh music

julius-cy contains scripts and documentation that make the whole process of installation very easy on a Linux based machine such as Raspberry Pi. This could be deemed difficult for a non-specialist given the complexity of the packages and configuration required for the acoustic models, pronunciation lexicons as well as any necessary grammar and vocabulary files. Further scripts make it possible for julius-cy to recognise users' own additions to the grammar and vocabulary files.

The open source project Jasper (Marsh and Saha, 2014) was used as our platform for applying julius-cy in a complete intelligent digital assistant solution. The persona name 'Macsen' was chosen as the wake up word that would instruct the Jasper system to alternate between passive and active listening modes. Further, but minimal, alterations were necessary to permit Jasper to support interaction in languages other than English. Modules written in simple Python were added to integrate various Welsh language websites such as Golwg360 News and S4C Weather. To voice responses to user requests 'Macsen' uses either a Welsh language Festival based text-to-speech voice (Prys et al., 2004) or the more naturally sounding 'Geraint' voice from Ivona's Speech Cloud.

5 Further Work and Conclusions

Much work remains on developing a Welsh language intelligent digital assistant. We are releasing all models, scripts, code and data developed by the project via the Welsh National Language Technologies Portal (Prys and Jones, 2015) as well as GitHub (Jones, 2016a; Jones, 2016b), according to the permissive MIT open source license which allows the widest possible outreach to other developers involved in commercial, education and volunteer activities for Welsh and other languages. We welcome all feedback and pull requests for extending its capabilities and improvements.

We intend for julius-cy to support large vocabulary continuous speech recognition by utilising the large text resources available to us, including the 30 million word Cysill Ar-lein corpus (Prys, Prys & Jones, 2016, forthcoming), to produce language models. This would allow opportunities for Macsen to support more domains and intelligent capabilities.

One aim is to investigate applying machine translation to translate texts recognized by julius-cy, from Welsh to

English in order to consume English medium APIs provided by wit.ai and/or SIRIUS. This allows Welsh speaking users to still use intelligent capabilities that are rooted in the English language.

The work on improving acoustic models in the meantime will continue. Additional funding was secured recently from Bangor University's Undergraduate Internship Scheme to employ a student to recruit and collect recordings from Welsh speaking staff and students in order to expand the amount of quality assured contributions.

Further work on acoustic and language modelling for Welsh are to be the subject of new KESS (Knowledge Economy Skills Scholarships) PhD programmes in partnership with the Welsh Government which will explore developing speech technologies using more recent developments in deep learning and neural network approaches.

Many in the Welsh language community recognise the opportunities and risks posed by the growing number of services and apps that provide intelligent capabilities via speech interfaces. Other similar language communities also recognise the benefit of creating language resources in this domain.

However, lesser resourced languages are required to be innovative in attracting funding. Certain funders desire useful end products and applications with wide reaching public impact. This skews longer term aims for research and development of basic language technologies and resources such as speech recognition for lesser resourced languages. There is a danger of users relying on widely available English language products and services which will impact upon language use, vitality and diversity.

We hope that this work contributes to further development of intelligent digital assistants for lesser resourced languages as well as stimulating developments in the wider industry.

6 Acknowledgements

The 'Seilwaith Cyfathrebu Cymraeg' project reported on in this paper was made possible with the financial support of the Welsh Government, through its Technology and Digital Media in the Welsh Language Fund, and S4C. The authors would also like to thank the contributors from various hackers and communities of users that assisted us on the project.

7 Bibliographic References

AonSquared. [No date]. *Speech recognition using the Raspberry Pi.* Available at: http://aonsquared.co.uk/raspi_voice_control [Accessed: 16 February 2016]

Cooper, S. Jones, D. B. and Prys, D. (2014) Developing further speech recognition resources for Welsh. In J. Judge, T. Lynn, M. Ward and B. Ó Raghallaigh (Eds.) *Proceeedings of the First Celtic Language Technology Workshop at the 25th International Conference on Computational Linguistics (COLING 2014),* pp. 55-59.

Cooper, S., Chan, D., Jones, D.B. (2015). *Corpws Lleferydd Paldaruo.* [http://techiaith.cymru/corpora/Paldaruo]

Ghazali, S. , Jones D. B., Prys D. (2015). *Towards a Welsh Language Intelligent Personal Assistant: A Brief Study of APIs for Spoken Commands, Question and Answer Systems and Text to Speech. Report for the Welsh Government*. Available at : http://techiaith.bangor.ac.uk/towards-a-welsh-language-intelligent-personal-assistant/?lang=en [Accessed: 23 March 2016]

Hauswald, J., Laurenzano, M. A., Zhang, Y., Li, C., Rovinski, A., Khurana, A., Dreslinski, R., Mudge, T., Petrucci, V., Tang, L. & Mars, J. (2015) Sirius: An Open End-to-End Voice and Vision Personal Assistant and Its Implications for Future Warehouse Scale Computers. In *Proceedings of the Twentieth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, New York, NY: ACM.

HTK. [No date]. *Hidden Markov Toolkit.* Available at: http://htk.eng.cam.ac.uk/ [Accessed: 16 February 2016]

Jones D.B. (2015) *Paldaruo – an iOS app for crowdsourcing speech data.* Available at https://github.com/techiaith/Paldaruo [Accessed: 16 February 2016]

Jones D.B. (2016a) *Macsen – A Welsh Language Intelligent Digital Assistant*. Available at https://github.com/techiaith/macsen [Accessed: 16 February 2016]

Jones D.B. (2016b) *An easy to use speech recognition development toolkit using HTK, Julius and Docker* Available at https://github.com/techiaith/seilwaith [Accessed: 23 March 2016]

Jones D.B. and Cooper, S. (2016) *julius-cy – Welsh language speech recognition with Julius,*Available at: https://github.com/techiaith/julius-cy [Accessed: 16 February 2016]

Jones D.B., Ghazali, S. (2016) *Project Seilwaith Cyfathrebu Cymraeg Website*. Available at http://techiaith.bangor.ac.uk/seilwaith-cyfathrebu-cymraeg/ [Accessed: 16 February 2016]

Marsh, C. and Saha, S. (2014) *Jasper Documentation*, Available at: http://jasperproject.github.io [Accessed: 16 February 2016]

MIT. [No date]. *The MIT License*. Available at: http://opensource.org/licenses/mit-license.html [Accessed: 16 February 2016]

Prys, D., and Jones D. B. (2015). National Language Technology Portals for LRLs: A Case Study. Paper presented at *Language Technologies in Support of Less-Resourced Languages, (LRL 2015)* 28 November 2015, Poznan, Poland.

Prys, D., Williams, B., Hicks, B., Jones, D., Ní Chasaide, A., Gobl, C., Carson- Berndsen, J., Cummins, F., Ní Chiosáin, M., McKenna, J., Scaife, R. and Uí Dhonnchadha, E. (2004). WISPR: Speech Processing Resources for Welsh and Irish. In Proceedings of the *Pre-Conference Workshop on First Steps for Language Documentation of Minority Languages*, LREC Conference, Lisbon, Portugal.

Prys, D; Prys, G; & Jones, D.B. (2016, forthcoming) Cysill Ar-lein: A Corpus of Written Contemporary Welsh compiled from an on-line Spelling and Grammar Checker. In *Proceedings of the 10th International*

Conference of Language Resources and Evaluation, LREC 2016. Portorož, Slovenia.

Welsh Government (2012). A living language, a language for living. Available at: http://wales.gov.uk/docs/dcells/publications/122902wls201217en.pdf [Accessed: 25 March 2016].

Welsh Government. (2013). Welsh language Technology and Digital Media Action Plan. Available at: http://wales.gov.uk/docs/dcells/publications/230513-action-plan-en.pdf [Accessed: 25 March 2016].

wit.ai [No date] *wit.ai: Natural Language for Developers.* Available at: https://wit.ai [Accessed: 16 February 2016]