Using IBM Watson Services to Process Video to Streamline Business Processes and Improve Customer Experience

Sarah Packowski
spackows@ca.ibm.com
IBM
Ottawa, Ontario, Canada

Rashmi Kulkarni
rashmi.kulkarni@ibm.com
IBM
San Jose, California, United States

Sharyn Richard
sharyn.richard@us.ibm.com
IBM
Boston, Massachusetts, United States

Gary Faircloth
gfaircl@us.ibm.com
IBM
Seattle, Washington, United States

ABSTRACT

Artificial intelligence (AI) technologies such as speech to text, text to speech, and natural language processing are now available through software-as-a-service offerings from multiple cloud service providers. The performance and accuracy of these technologies continue to improve. However, it can still be a struggle to build end-to-end infrastructure and automation using these technologies to support business processes and goals. In this paper, we describe our team's experience using IBM Watson services on IBM Cloud to process internal meeting recordings and external product videos. We create accurate, well-formatted text transcripts and captions files, implement video transcript search, and generate translated audio. These solutions improve our internal processes and provide a better experience for our customers.

CCS CONCEPTS

• **Information systems** → **Video search**; • **Applied computing** → *Enterprise computing*; • **Human-centered computing** → *Accessibility*.

KEYWORDS

IBM Watson, Speech to Text, Text to Speech, Video search

ACM Reference Format:

Sarah Packowski, Rashmi Kulkarni, Sharyn Richard, and Gary Faircloth. 2021. Using IBM Watson Services to Process Video to Streamline Business Processes and Improve Customer Experience. In *Proceedings of 30th Annual International Conference on Computer Science and Software Engineering (CASCON '21).* ACM, New York, NY, USA, 6 pages.

MOTIVATION

This paper describes our experience using Watson services on IBM Cloud to accomplish two business goals:

(1) Get more value from our internal meeting recordings.
(2) Provide a better product video experience for our customers.

CASCON '21, November 2021, Toronto, Ontario Canada

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.

Internal meeting recordings

Because our team employs Agile practices that prioritize real-time discussion, we share a lot of information in meetings. Our team is globally distributed, we use web conference tools to facilitate meetings, and our meetings are conducted in English.

- **Challenges for individuals** - A team member might miss a meeting. When they attend, they might be unable to hear what is said or unable to follow the discussion. A team member who misses important information is at a disadvantage.
- **Challenges for organizations** - Time is wasted when multiple meetings are needed to propagate information from one meeting group to another or when multiple groups are unknowingly discussing the same blocking issues.

These challenges can be addressed by generating captions in real time and by recording meetings and then making the transcripts available to be reviewed and searched. Legal, business, and privacy implications must be considered: not all meetings should be recorded, not all team members wish to be recorded, and not all team members have a business need to access all recordings.

External product videos

Our customers tell us they want product information in video format. They want videos that explain concepts, demonstrate tasks, and show troubleshooting steps. We produce hundreds of videos for each of our products. For one of our products, for example, the number of product videos grew by 800% over the past five years.

The serial nature of the video format poses a challenge. Nobody wants to watch a video without first knowing it has the information they want. Customers have told us that finding the right video and then finding the information they need in that video is difficult.

Navigating textual information doesn't have to be linear:

- You can search for a phrase and then click a search result to jump right to where that phrase is mentioned.
- You can skim topic titles, page headings, and the first sentence in each paragraph to get a sense for a topic.
- You can read just the paragraph or instruction that you need without having to read anything else.

The same user experience can be provided for video by publishing video transcripts in product documentation. This way, the transcript text can be searched and skimmed, and customers can watch just the 10 seconds of video they need.

PROJECTS

This paper describes our experience with six video-related projects:

(1) **Transcript pipeline** - Produce accurate transcript files in multiple formats for use in other projects.
(2) **Model maintenance** - Automatically evaluate and retrain our speech-to-text and punctuation-restoration models.
(3) **Meetings summary** - Build a web app that provides highlights from internal meetings and a search feature for finding specific topics in meeting discussions.
(4) **Videos in product documentation** - Embed videos in product documentation, complete with transcript text.
(5) **Video search** - Be able to search videos based on titles, chapters, transcript text, scene descriptions, and more.
(6) **Video translation** - Publish video transcript text, on-video captions, and audio in multiple languages.

Samples demonstrating methods described in this paper are here: https://github.com/spackows/CASCON-2021_Processing_video

PROJECT 1: TRANSCRIPT PIPELINE

The goal of Project 1 is to produce the transcript files required by all subsequent projects listed in this paper.

Stage 1: Collect or generate an initial, imperfect transcript.

- **Real-time, automated captions** - Our internal meeting recordings sometimes have a transcript generated by the web conference software used to host the meeting.
- **Speech to text from recordings** – For the rest of our videos, we generate a transcript using IBM Watson Speech to Text with a model customized for product-specific terminology.

Even state-of-the-art speech recognition produces many errors for some speakers [4]. Also, in meeting recordings, speakers might trail off, be interrupted, or use a lot of domain-specific jargon. In fact, direct speech-to-text transcripts of meeting recordings is arguably *neither speech nor text* [10]. For all of these reasons, this initial transcript must be further processed to be useful.

Stage 2: Automatically process initial transcript.

(1) **Restore punctuation** – If the initial transcript doesn't have punctuation, we add punctuation using a model built with the Python library `punctuator` [1].
(2) **Detect sentence boundaries** – Identify where sentences begin and end using the Natural Language Toolkit, `NLTK`[2].
(3) **Set caption breaks** – We want each caption to include complete sentences, because this sets every other project up for success. So, we break the text into captions at the sentence level and then adjust timestamps.

Stage 3: Manually correct and validate transcript text.

(1) **Segment video** - The original video and the transcript from Stage 2 are divided into short segments (about 2.5 minutes long, depending on where a sentence break occurs.)
(2) **Correct each segment** - Teammates sign into a web app and then select a segment to manually correct and validate. The web app shows the video segment in a player and the corresponding transcript segment is presented for editing.
(3) **Assemble full transcript** - After all segments are corrected, they are assembled into a complete, correct transcript.

[1]https://pypi.org/project/punctuator
[2]https://www.nltk.org

Table 1: Example output from Project 1 transcript pipeline stages 1-3

Stage	Output	Notes

Stage 1	`00:01:15`
	`next you will receive an email with a seven digit verification code copy the code from the email`
	`00:01:24`
	`and pasted on the registration form`
		Mistakes:
		• There is no punctuation.
		• Multiple sentences are included in the first caption.
		• The compound adjective "seven digit" is not hyphenated.
		• "pasted" appears instead of the phrase "paste it".

Stage 2	`00:01:15`
	`Next, you will receive an email with a seven digit verification code.`
	`00:01:20`
	`Copy the code from the email and pasted on the registration form.`
		Improvements:
		• Proper punctuation and capitalization.
		• Each caption contains one sentence.
		• The timestamp of the second caption has been adjusted based on the word timestamp associated with the first word in the second sentence, "Copy".

Stage 3	`00:01:15`
	`Next, you will receive an email with a seven-digit verification code.`
	`00:01:20`
	`Copy the code from the email and paste it on the registration form.`
		Corrections:
		• A hyphen added to "seven-digit".
		• "pasted" has been corrected to "paste it".

Manually transcribing one hour of video into text can take four to five hours [3] and is tedious work. However, correcting a short segment in our transcript-correction tool takes less than ten minutes, which a team member can easily fit in here and there, so our team can share the work.

Some research suggests that engaging with course-related videos, by adding annotations or identifying errors for example, might help students learn course material [14], [15]. We believe correcting meeting transcripts - including listening to a video segment more than once to confirm technical jargon or searching for the correct spelling of a domain-specific term - helps new team members learn about team processes and projects.

Stage 4: Generate transcript output files.

- **Plain text files** - For use with natural language processing, such as to summarize a recorded meeting for Project 3.
- **Captions files** - In VTT format, for use with all videos, wherever they are hosted.
- **HTML** - For including in our meetings app for Project 3 and in our product documentation for Projects 4 and 6.
- **JSON** - Enables searching transcripts for Projects 3 and 5.

PROJECT 2: MODEL MAINTENANCE

The goal of Project 2 is to leverage the corrections from the transcript pipeline in Project 1 to automate the improvement of our speech-to-text and punctuation-restoration models.

Retrain speech-to-text model

For transcripts that were generated using Watson Speech to Text, we regularly evaluate which words or phrases the model is getting wrong. We do this by comparing the final, correct transcript from Stage 4 of the transcript pipeline from Project 1 with the initial transcript from Stage 1. At regular intervals, we collect these *words lists* and then update our customized Watson Speech to Text model.

Table 2: Transcript with domain-specific terminology

Stage	Output
Stage 1	we need to expand the sand pile system
Stage 4	We need to expand the SAN file system.

Table 2 shows transcript text that refers to a storage area network, often called by its acronym, SAN, pronounced like a word: "san". When we run a *difference algorithm*[3] to compare the initial transcript with the corrected transcript, we can see the mistake "sand pile system" has been corrected to "SAN file system". From this difference, we would add "SAN file system" to our words list to customize our speech-to-text model.

Retrain punctuation-restoration model

To perform punctuation restoration, we use the Python library `punctuator`. We initially created one model, trained on meeting recordings from the International Computer Science Institute (ICSI) data set [9]. However, although that model works well for our meeting recordings, it doesn't work well for our product videos.

[3]https://www.npmjs.com/package/diff

Table 3 shows some transcript text from a product video with punctuation added by our model trained on the ICSI data set (first row) compared with the correct punctuation added manually (second row.) The list of formal product names is just not like something someone would have said in the ICSI meeting recordings.

Table 3: Punctuation model trained on ICSI data set

Method	Result
Model	`Here. You see the Watson services that will be provisioned, Watson, Studio, Watson, Machine Learning and Watson Knowledge, Catalog.`
Manual	`Here, you see the Watson services that will be provisioned: Watson Studio, Watson Machine Learning, and Watson Knowledge Catalog.`

Because the language in our meeting recordings is so different from the language in our product videos, we created two models:

- **For punctuating meeting recordings** - Model 1, trained on meeting recordings from the ICSI data set.
- **For punctuating product videos** - Model 2, trained on manually corrected transcripts of our product videos.

For transcripts that had punctuation added by one of our models, we regularly evaluate punctuation that was missed or added incorrectly. We do this by comparing the final, correct transcript from Stage 4 of the transcript pipeline from Project 1 with the output from Stage 2. For transcripts with many errors, we add the corrected transcript to the training data and then retrain the model.

PROJECT 3: MEETINGS SUMMARY

The goal of Project 3 is to make better use of internal meeting recordings, to turn them into artifacts that help the team work.

Summary

We wanted to build more than a meeting transcripts searching tool. We wanted to summarize [12], [7] and highlight key meeting details too. Here are three types of highlights we chose:

- **Agenda** - We classify sentences in the transcript text by theme, based on keywords. Those themes form the agenda.
- **Questions** - We use question marks to identify questions.
- **Follow-up items** - To capture promises made to do something after a meeting, we look for sentences in future tense.

About wake words. At first, we tried to have meeting participants adopt special phrases, such as "agenda item" or "to do". But it felt awkward and people struggled to remember to use them. Because of the popularity of voice-activated devices, we thought people would be familiar with using a wake word, but there is evidence people struggle with that in other contexts too [1].

Reducing noise. The simple techniques listed above capture questions such as "Can you see my screen?" and follow-up items such as "That will be great!" Reducing this noise is an active area of work.

Web app

We created a web app that team members can use to search internal meetings, and then easily watch the video recording and read the full transcript. The app lists the highlights for each meeting, and allows team members to manually add agenda items, questions, follow-up items, and general notes. The app is also integrated with collaboration tools, such as GitHub and Slack.

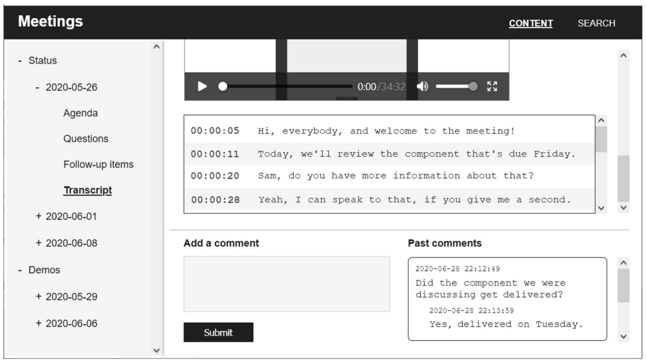

Figure 1: Meetings app

PROJECT 4: VIDEOS IN PRODUCT DOCUMENTATION

The goal of Project 4 is to make it as easy as possible for all users to find and understand conceptual, how-to, and troubleshooting information in our external product videos.

On web pages containing text, readers use information foraging strategies such as searching for keywords, skimming headings, non-linear reading, and reading selectively to get just the information they need [6].

People who use screen readers skim by using macro- and micro-structures such as table of contents, headers, and links built into the document or by employing strategies such as reading only the first sentence of each paragraph [13].

Many factors combine to make dense, technically complex information difficult to understand when presented in traditional, on-video captions [11]:

- Mistakes, particularly with domain-specific terminology.
- Lack of punctuation and capitalization.
- Only a few words displayed on the screen at once.
- Timing of captions too fast, too slow, or just unnatural.
- No easy way to quickly review what was previously said.
- Reading captions while watching video in peripheral vision.
- Poor contrast between background and caption text.
- Captions can obscure content in the video.

In additional to these factors, for anyone reading these captions in a language that is not their first language, or who is learning concepts, terms, and technology for the first time, understanding is even more difficult.

Figure 2 shows our attempt to provide an improved video experience: topics in the product documentation that embed a video and include transcript text in an interactive element just below the video player.

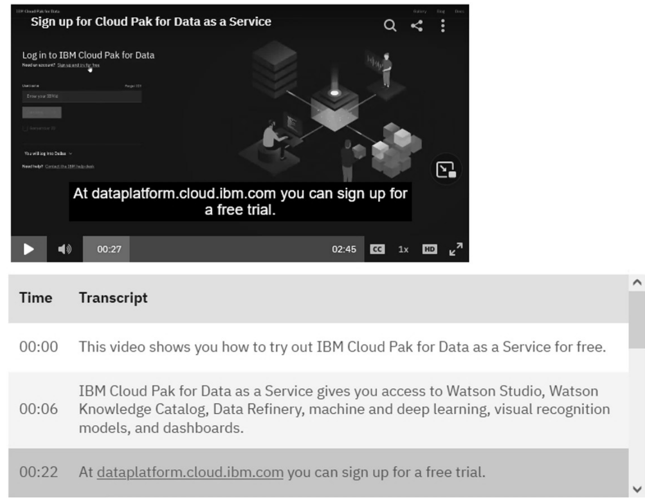

Figure 2: Example documentation topic

Design choices made to meet the goal of Project 4:

- On-video captions are complete, punctuated sentences and remain on screen longer.
- Full transcript text, formatted for easy reading, is provided in a collapsible element just below the video player.
- The transcript is positioned to minimize how far the eye must travel between viewing the player and reading the text.
- A visual cue (darker background) highlights the sentence relevant to the current point in the video, making it easy for the eye to jump to the correct sentence.
- The transcript table automatically scrolls to keep the text in view as the video plays.
- The transcript table enables a screen reader to jump from one sentence to the next easily.
- Clickable timestamps enable selectively watching parts of the video or watching in a non-linear way.

PROJECT 5: VIDEO SEARCH

The goal of Project 5 is to be able to search in video transcript text as easily as we can search in HTML web pages.

Our implementation uses JSON-formatted metadata files, produced by the transcript pipeline from Project 1, with a search engine like Elasticsearch or with IBM Watson Discovery. Figure 3 shows a basic example.

In addition to an array of timestamped captions, the JSON structure gives us the flexibility to include arrays of video chapters, scene descriptions, on-screen text extracted using visual recognition, background noise, or any other property we can imagine searching on.

Whether it's our meetings app from Project 3, or in our product documentation, we can present video search results as shown in Figure 4. In these video search results, you can view the video from the beginning by clicking the video title, or you can view the video from a specific timestamp by clicking one of the matching captions.

```
{
  "title" : "Signing up for a personal account"
  "url" : "https://dataplatform...signup-wdp.html"
  "captions" : [
    ...
    {
      "seconds" : 75,
         "caption" : "Next, you will receive an email with
         a seven-digit verification code."
    },
    ...
  ]
}
```

Figure 3: Example video search metadata

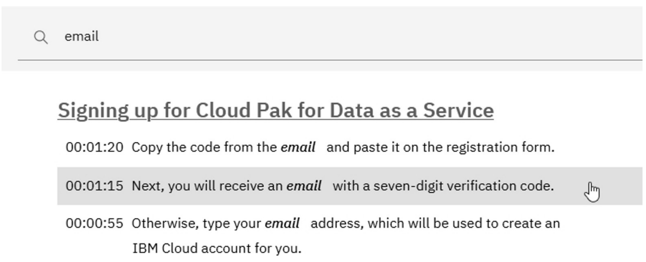

Figure 4: Example video search results

PROJECT 6: VIDEO TRANSLATION

The goal of Project 6 is to translate external product videos: transcript text in product documentation, on-video captions, and audio.

Transcript text in product documentation

Our team has an established process for translating product documentation. For Project 6, we use the existing process to translate product documentation topics that embed videos, including the table containing the transcript text (from Project 4):

(1) English HTML transcript table row:

```
<tr>
<td>01:15</td>
<td>Next, you will receive an email with a seven-digit
verification code.</td>
</tr>
```

(2) Translated (French) HTML transcript table row:

```
<tr>
<td>01:15</td>
<td>Vous allez recevoir un e-mail avec un code de
vérification à sept chiffres.</td>
</tr>
```

Remember that in Stage 2 of Project 1, we set caption breaks based on sentence boundaries. This ensures each row of the transcript table in the documentation topic contains a complete sentence. This is important for translation because it is difficult for translators to translate sentence fragments well.

On-video captions

From the translated documentation topics, we generate translated on-video captions files in VTT format. Each row of the transcript table is one caption:

(1) French transcript table (HTML):

```
<tr>
<td>01:15</td>
<td>Vous allez recevoir un e-mail avec un code de
vérification à sept chiffres.</td>
</tr>
<tr>
<td>01:20</td>
```

(2) French on-video caption (VTT):

```
00:01:15.100 --> 00:01:19.900
Vous allez recevoir un e-mail avec un code de
vérification à sept chiffres.
```

Audio

We use IBM Watson Text to Speech to synthesize audio for each of the translated captions. (Here again, this is why it's so important for each caption to contain complete sentences.) To produce the final video with translated audio, we join up the audio from the translated captions, aligned with the caption timestamps.

For our product videos, it is common for translated audio to be longer than the original English audio. Figure 5 shows the average length of translated audio captions for the videos in our project, as a percentage of the English audio for the same captions.

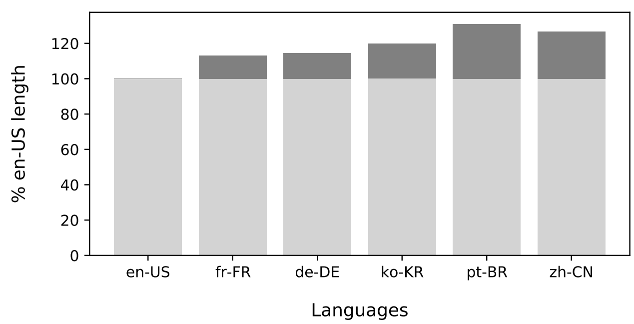

Figure 5: Audio translation expansion

When the translated audio is longer than the English audio, we must adjust. We are experimenting with two methods:

- Method 1: Adjust caption timestamps.
- Method 2: Add frames to the video.

Method 1: Adjust caption timestamps. If the translated audio is too long for only a few captions in a video, we can shift some caption timestamps slightly earlier or slightly later to make all the audio fit.

For example, Table 4 lists the start times and duration (in seconds) of the audio for three captions in English and French.

Figure 6 shows the same information visually. The French audio for the first caption is so long that it overlaps the original start time of the second caption. Because the third caption doesn't start until much later, when assembling the full, French audio, we can delay the start of just the second caption to make things fit.

Table 4: Sample caption timings

Caption	Start time	Duration (en-US)	Duration (fr-FR)
1	0	5.3	6.3
2	6	11.5	12
3	22	8.2	8.4

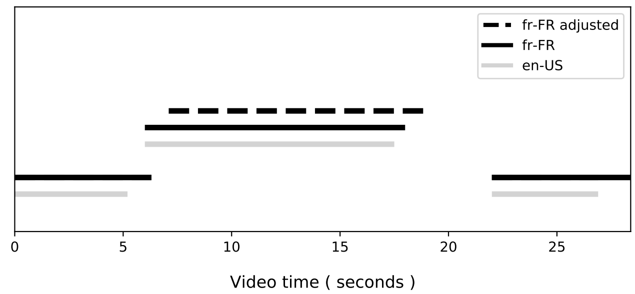

Figure 6: Shifting the timestamp of the second caption

Method 2: Add frames to the video. Another way to solve the previous timing problem is to slow down the visuals to add time for the longer audio. For example, for the first 6 seconds of the video, duplicating every fifth frame would cause that section of video to last 7.2 seconds, which is long enough to fit the first French caption audio. We would then delay all subsequent French caption start times by 1.2 seconds.

CONCLUSION

Developing a project from a proof-of-concept demonstration to reliable business process infrastructure is a challenge for AI projects, whether they're analytics and business intelligence projects [5] or projects like the ones listed in this paper.

Two things have helped our team get past the proof-of-concept stage into deployed, productive solutions:

- **Using AI software as a service** - We can rapidly prototype solutions because we don't have to start with building and training models from scratch by hand. We can use default models at first, and then customize them. Finally, We can easily substitute different cloud services to experiment [2].
- **Approaching projects with a platform mindset** - Designing components as web services with APIs enables other projects to experiment and build on top [8]. In the projects described in this paper, you can see how Project 1 is the foundation which makes all the other projects possible, and you can see how Project 4 enables Projects 5 and 6.

ACKNOWLEDGMENTS

The authors would like to thank Hai-Nhu Tran, Program Director, Content Design for IBM Data and AI, as well as Jason Shayer, Wendy Switzer, and Brandon Elkins for their support and encouragement.

REFERENCES

[1] Shashank Ahire and Michael Rohs. 2020. Tired of Wake Words? Moving Towards Seamless Conversations with Intelligent Personal Assistants. In *Proceedings of the 2nd Conference on Conversational User Interfaces* (Bilbao, Spain) *(CUI '20)*. Association for Computing Machinery, New York, NY, USA, Article 20, 3 pages. https://doi.org/10.1145/3405755.3406141

[2] Pablo Ángel Álvarez Fernández and Jeremy R. Hajek. 2020. A Case Study in Comparative Speech-to-Text Libraries for Use in Transcript Generation for Online Education Recordings. In *Proceedings of the 21st Annual Conference on Information Technology Education* (Virtual Event, USA) *(SIGITE '20)*. Association for Computing Machinery, New York, NY, USA, 223–228. https://doi.org/10.1145/3368308.3415380

[3] Christopher Downey Christian Bokhove. 2018. Automated generation of 'good enough' transcripts as a first step to transcription of audio-recorded data. https://doi.org/10.1177/2059799118790743

[4] Siyuan Feng, Olya Kudina, Bence Mark Halpern, and Odette Scharenborg. 2021. Quantifying Bias in Automatic Speech Recognition. arXiv:2103.15122 [eess.AS]

[5] Iris Figalist, Christoph Elsner, Jan Bosch, and Helena Holmström Olsson. 2020. Breaking the Vicious Circle: Why AI for software analytics and business intelligence does not take off in practice. In *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. 5–12. https://doi.org/10.1109/SEAA51224.2020.00013

[6] Gemma Fitzsimmons, Mark J. Weal, and Denis Drieghe. 2014. Skim Reading: An Adaptive Strategy for Reading on the Web. In *Proceedings of the 2014 ACM Conference on Web Science* (Bloomington, Indiana, USA) *(WebSci '14)*. Association for Computing Machinery, New York, NY, USA, 211–219. https://doi.org/10.1145/2615569.2615682

[7] Pei-Yun Hsueh and Johanna D. Moore. 2009. Improving Meeting Summarization by Focusing on User Needs: A Task-Oriented Evaluation. In *Proceedings of the 14th International Conference on Intelligent User Interfaces* (Sanibel Island, Florida, USA) *(IUI '09)*. Association for Computing Machinery, New York, NY, USA, 17–26. https://doi.org/10.1145/1502650.1502657

[8] Thierry Isckia, Mark de Reuver, and Denis Lescop. 2018. Digital Innovation in Platform-Based Ecosystems: An Evolutionary Framework *(MEDES '18)*. Association for Computing Machinery, New York, NY, USA, 149–156. https://doi.org/10.1145/3281375.3281377

[9] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, Nathaniel Morgan, B. Peskin, Thilo Pfau, Elizabeth Shriberg, A. Stolcke, and Chuck Wooters. 2003. The ICSI meeting corpus. I–364. https://doi.org/10.1109/ICASSP.2003.1198793

[10] Raja S. Kushalnagar, Gary W. Behm, Aaron W. Kelstone, and Shareef Ali. 2015. Tracked Speech-To-Text Display: Enhancing Accessibility and Readability of Real-Time Speech-To-Text. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility* (Lisbon, Portugal) *(ASSETS '15)*. Association for Computing Machinery, New York, NY, USA, 223–230. https://doi.org/10.1145/2700648.2809843

[11] Raja S. Kushalnagar, Walter S. Lasecki, and Jeffrey P. Bigham. 2013. Captions versus Transcripts for Online Video Content. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility* (Rio de Janeiro, Brazil) *(W4A '13)*. Association for Computing Machinery, New York, NY, USA, Article 32, 4 pages. https://doi.org/10.1145/2461121.2461142

[12] Fei Liu and Yang Liu. 2008. What Are Meeting Summaries? An Analysis of Human Extractive Summaries in Meeting Corpus. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue* (Columbus, Ohio) *(SIGdial '08)*. Association for Computational Linguistics, USA, 80–83.

[13] Tonja Machulla, Mauro Avila, Pawel Wozniak, Dillon Montag, and Albrecht Schmidt. 2018. Skim-Reading Strategies in Sighted and Visually-Impaired Individuals: A Comparative Study. In *Proceedings of the 11th PErvasive Technologies Related to Assistive Environments Conference* (Corfu, Greece) *(PETRA '18)*. Association for Computing Machinery, New York, NY, USA, 170–177. https://doi.org/10.1145/3197768.3201535

[14] Abelardo Pardo, Negin Mirriahi, Shane Dawson, Yu Zhao, An Zhao, and Dragan Gašević. 2015. Identifying Learning Strategies Associated with Active Use of Video Annotation Software. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge* (Poughkeepsie, New York) *(LAK '15)*. Association for Computing Machinery, New York, NY, USA, 255–259. https://doi.org/10.1145/2723576.2723611

[15] Oleksandra Poquet, Lisa Lim, Negin Mirriahi, and Shane Dawson. 2018. Video and Learning: A Systematic Review (2007–2017). In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge* (Sydney, New South Wales, Australia) *(LAK '18)*. Association for Computing Machinery, New York, NY, USA, 151–160. https://doi.org/10.1145/3170358.3170376