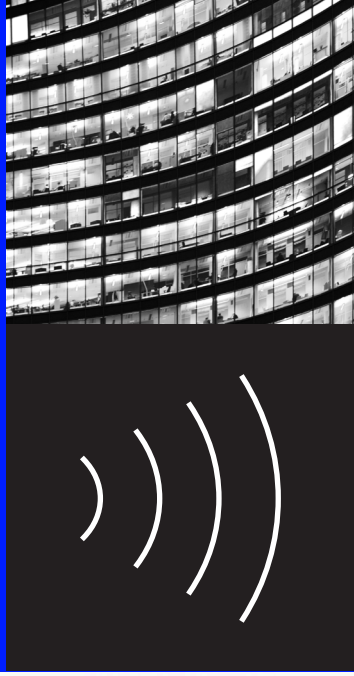
Using Automatically Created Confidence Measures

in the Transcription of Financial Events



LSEG DATA & ANALYTICS

Table of contents

LSEG DATA & ANALYTICS

Executive Summary

As part of the London Stock Exchange Group (LSEG), LSEG Data & Analytics provides approximately 40,000 transcripts of financial events per year[1] which are produced by our Data & Analytics Content Operations team, a group of highly skilled domain experts. Given the vast volume of events that are covered, and the rigorous standard of quality required, we consistently strive to develop innovative measures for increasing the efficiency and accuracy of the workflow used by our transcripts production team. One such development has been the use of automatically created confidence measures, a project which has leveraged machine learning.

Machine learning is used for a multitude of natural language processing (NLP) tasks. As the use of automatically generated predictions has become increasingly pervasive in many scenarios, the ability to assess the "correctness" of these predictions has become a dedicated research area of its own. Such measures of correctness represent the confidence that a system has in the decisions it makes, and they have wide-ranging implications in many diverse machine learning areas from self-driving vehicles to medical diagnosis.

In the domain of automatic speech recognition (ASR), confidence measures represent the probability that the output text is the correct transcript of what was said. These confidence measures can be generated on a variety of granularities, from the sub-word level to the document level. There are a variety of different ways of retrieving such measures: the ultimate solution will depend on the data available, the machine learning architecture being used, and the downstream use-case.

The work outlined in this paper describes the research work that was undertaken by LSEG's in-house speech processing experts, the Centre of Expertise in Spoken Language Technologies (CESLT) to provide confidence measures for the output of their ASR system. This work was conducted as part of a broader project which saw the establishment of a pipeline for automatically transcribing financial events. This automatically generated output is manually corrected by our domain experts in a purpose-built UI. By providing document-level confidence measures, the resource allocation of work was made more efficient.

In this paper, we discuss the research that was conducted into this capability. Additionally, we outline some of the experiments that were carried out and how their results shaped the project. Finally, we discuss some of the wider implications of this project, in addition to some suggested next steps.

[1] LSEG Data & Analytics: Transcripts & Briefs

1. Background

1.1 Use-Case: The Transcripts Project

In this paper, we describe work undertaken under the Transcripts project, the aim of which was to bring the production of transcripts for corporate events and earnings calls in-house, replacing the existing external vendor. Within this new workflow, ASR is used to transcribe the speech in both recorded and live corporate events and earnings calls to create an initial draft transcript. During the automatic transcription of an audio recording, statistics are gathered and converted into confidence measures. These confidence measures are then propagated into ASR output. Finally, the draft of the transcript is edited into a final form by LSEG's highly skilled team of domain experts, before being published to LSEG Workspace.

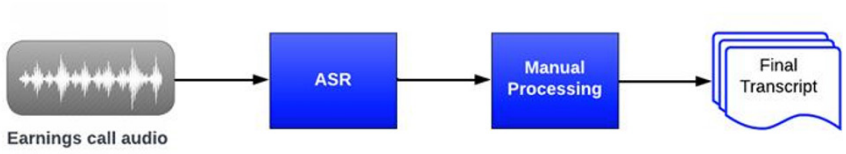


Figure 1: New, updated workflow for the transcription of financial events and earnings calls. ASR has replaced functionality that was being provided by a third-party vendor. Source: LSEG, as of November 22, 2024.

Using ASR in conjunction with the established expertise from our domain experts allows LSEG to significantly increase coverage by bringing the full operation in-house. The operation continues to provide high quality output which can further be enhanced and enriched with existing natural language processing capabilities developed by our Applied NLP team, such as summarization, sentiment analysis and entity tagging.

1.2 Automatic Speech Recognition

Automatic Speech Recognition (ASR) automatically converts speech into text using statistical models. Broadly speaking, the architectures used for ASR can be split into two categories:

- Systems which use two separate statistical models in combination: one which predicts the acoustic units from the recording, i.e., an acoustic model (AM), and another which uses these predicted acoustic units to predict strings of characters, i.e., a language model (LM), so-called "factorised" systems.
- Systems which model strings of characters from the audio explicitly, so-called "end-to-end" systems.

CESLT explored the use of confidence measures under both architectures, however it is the former that will be referenced in this paper.

At the start of this project, CESLT made use of a speech recognition toolkit provided by Kaldi[2]. Kaldi formed the skeleton of our pipeline, with which we have built various adaptations, and trained various composite models. The underlying structure used by this toolkit is a lattice [1], a data structure which stores all alternative hypotheses labelled with probabilities and timing information during recognition.

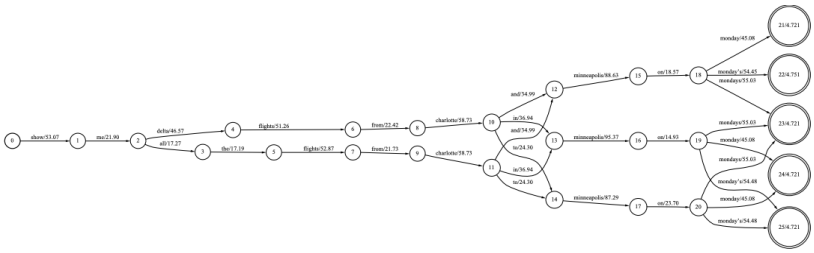


Figure 2: A lattice for the phrase "Show me all the flights from Charlotte to Minneapolis on Monday." Source [1].

Multi-pass search is used to get the final hypothesis, combining decoding and rescoring "passes" to refine and re-refine the output (Figure 3). During each decoding pass, the following steps are taken:

- We create a lattice of arbitrary width using context-dependent phonetic information. The lattice can contain both the acoustic and language model likelihoods.
- We iteratively refine the hypothesis using Minimum Bayes Risk decoding [2] and calculate the posteriors from the highest probability sequence.

The final module in this multi-pass pipeline involves the rescoring of hypotheses from a pruned selection of the n-best possible hypothesis paths through the lattice. Confidence scoring approaches that rely on probabilities from such pruned structures can suffer from over-estimation, a problem we examine further in sections 2.1 and 2.2.

Figure 3: The three "passes" made during CESLT's decoding process. The first step involves a decode which is then rescored using a more complex model. The final rescoring step involves use of a recurrent neural network (RNN) LM on a set of the n-best hypotheses. The highest scoring path is retrieved. Source: LSEG, as of November 22, 2024.

Evaluation

Common metrics for evaluating ASR are word error rate (WER) and character error rate (CER). Both metrics require having a true record of what was said, or a "reference". These metrics measure the ratio of incorrectly hypothesised words/characters to correctly hypothesised words/characters out of all the words that were spoken, as shown in the equations below.

$$WER = \frac{substitutions + insertions + deletions}{total\ \#\ words\ in\ reference} * 100 \qquad CER = \frac{substitutions + insertions + deletions}{total\ \#\ characters\ in\ reference} * 100$$

Which evaluation metric to use depends on what needs to be evaluated: while WER evaluation is typically applied on unformatted, unpunctuated text, CER can score the punctuation, formatting, and capitalisation of all words in the hypothesis. This is particularly useful if you wish to analyse performance on items such as company names, acronyms, and initialisms. For the sake of the experiments outlined in section 2, we explore the use of both metrics.

[2] Kaldi ASR (kaldi-asr.org)

1.3 Confidence Measures

In the domain of ASR, a confidence score is generally taken to be the probability that a word in the hypothesis is correct. While ASR systems output probabilities representing the likelihood of an outcome, they can also optionally output confidences which measure *the degree of certainty in that probability.*

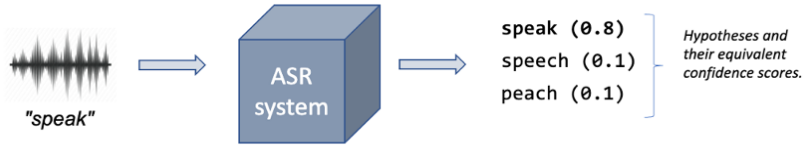


Figure 4: An input audio segment being fed through a generic ASR system. The multiple possible hypotheses with their equivalent confidences are made available. Source: LSEG, as of November 22, 2024.

Confidences can be created from several features and can broadly be split into categories of information sources as shown in the table below. The choice of what features to use will depend on the behaviour and architecture of the system that is being used.

Feature	Details (not exhaustive)
Language related	• LM scores (per word / per sentence) • Semantic information • Word frequency
Acoustic related	• AM scores (per frame / per phone)
Density/ratio related	• Number of alternative outputs spanning a certain predetermined window • Log likelihood ratio • Entropy

Several potential use-cases for confidence measures were explored as part of the Transcripts project, but the main focus was resource-management and enhanced user experience for our domain experts who perform the manual processing step as outlined in Figure 1. With the new confidence scoring capability, domain experts would be provided with a confidence label per-event, alongside the ASR output, as demonstrated in Figure 4. With this information, domain experts can triage events as well as predict editing effort and time required for each transcript. Armed with this information, resource allocators can be more effective in how they distribute work.

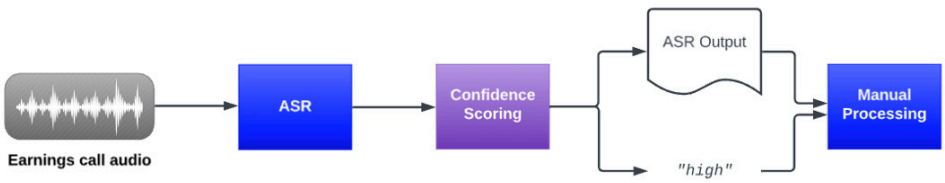


Figure 5: Transcripts workflow with enhanced confidence scoring capability. Each document now has an automatically assigned confidence label. Source: LSEG, as of November 22, 2024

With this use-case in mind, an extensive overview of market and academic approaches to producing confidence measures was conducted. From this, it was found that approaches could be divided into two categories: the first being those which use existing information, and the second being those which create a purpose-built model. For the former, "existing information" refers to statistics and figures which are calculated during the ASR decoding process. These statistics could refer to features such as the probability of a certain word. Systems which use these statistics will benefit from the efficiency of using such "by-products" from the decode. On the other hand, systems which are created separately from the decode allow for greater control over the input features and are more adaptable to the specific scenario. There is also the ability to combine both types of features in a hybrid approach. A broad summary of the pros and cons of both approaches can be found below.

	Existing information	Purpose-built model
Pros	Computationally cheap to gather confidence scores if they have been taken from pre-made values.	Greater control over what input features are used. Possibility for creating a system that could transfer between different ASR models.
Cons	Calculation might take place on a pruned sub-space which means that confidences could be overestimated. Pre-made values are tied to the specific ASR system that they were taken from.	More engineering effort. More compute power required. Requires training data which in this case would have to be documents labelled with confidences.

1.4 Data

We used datasets containing earnings calls and financial events that are highly representative of the domain and distribution of data in the Transcripts workflow. A portion of the data was a manually created ("gold-standard") representation of the content and timings of what was said in the audio. The creation of manually edited data can be an expensive task, so most of the data was formed of published events transcripts that had already been edited and formed our "large" dataset. As such, the content of these files could differ slightly from what was said in the audio since disfluencies and repetitions are removed. The alignment between text and audio was done automatically.

Dataset	Number of Files	Notes
"Large"	1007	Containing published Transcripts files which may not be entirely verbatim representations of what was said in the audio. The alignment of the text to audio was done automatically using CESLT's own segmentation model.
"Small"	74	Containing shorter sections of Transcripts files which have been manually aligned to the audio and edited to reflect what was truly said in the audio.
"Validation"	20	Of the same type as "large" dataset but provided later and used as the final validation set.

All datasets were chosen to be representative of the many varied event types, regions, and industries that are currently processed by the Transcripts workflow.

2. Exploration

2.1 Word-level Confidence Scores from Lattices

CESLT uses lattices as the underlying data structure during decoding. The first section of work that was conducted was to examine how confidences from this data structure could be calculated.

Using a gold-standard evaluation set, we can categorise the words in our output into words that were correctly hypothesised or words which were incorrectly hypothesised. Within the latter category we can separate the words into substitutions, deletions or insertions. Although we can gather the confidence for words which were substitutions or insertions, we cannot gather confidence statistics for deletions: we only have confidence scores for words which appeared in the hypothesis. For the words we do have, we were able to draw conclusions from the mean confidences scores which are demonstrated in the table below.

Multi-pass stage	Correct / Incorrect	% of Total Tokens	Mean Confidence
Rescore	Correct	92.6	0.97
	Incorrect	7.4	0.71
RNN LM Rescore (n-best)	Correct	91.7	0.99
	Incorrect	8.3	0.88

We were able to conclude that the confidences of correctly hypothesised words tend to be higher than those which are incorrectly hypothesised. As suspected, the mean confidence of incorrectly hypothesised words is still high (0.71 and 0.88), confirming our suspicion that these were over-estimated. Additionally, our suspicions that the n-best pruning in the final multi-pass step contributes to this overestimating by decreasing the gap between correctly and incorrectly hypothesised words' mean confidence scores. We decided to see if this over-estimation is demonstrated when we aggregate these values to the document-level, as discussed in the following section.

2.2 Document-level Confidences from Lattices

Using an aggregated confidence score per session, we decided to check if session-level error rate and session-level confidence were correlated. Using both the “large” and “small” evaluation sets as mentioned in Section 1.4, we created correlation graphs for each case. As outlined in Figure 3, we pass the decoded output into two consecutive rescore passes. As anticipated, the relationship between confidence and error was roughly linear up until the last rescore where the pruning results is a heavily overestimated confidence score. If we were to increase the number of paths that this pass was performed on, we would likely incur speed and efficiency losses. As such, it was decided that we would use the confidences from the first rescore module instead. Results for this module are outlined in the diagrams below.

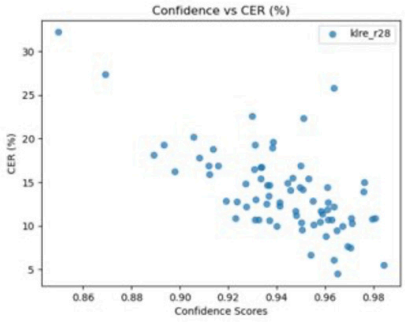


Figure 6: Aggregated confidence vs CER on the “small” evaluation set. There is a roughly linear relationship between the variables, however there are some outliers. Source: LSEG, as of November 22, 2024.

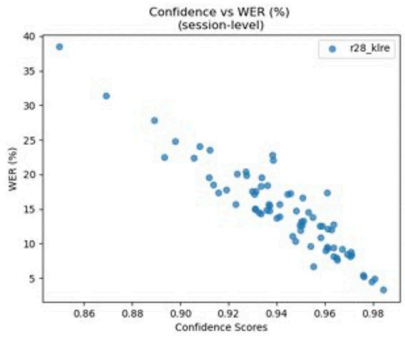


Figure 7: Aggregated confidence vs WER on the “small” evaluation set. The relationship between WER and confidence appears stronger, with less variance than the equivalent CER results outlined in Figure 6. Source: LSEG, as of November 22, 2024.

Both the CER and WER results show a negative correlation with the aggregated confidence score, however the relationship between WER and confidence is more pronounced, with significantly less variance.

Next, we performed the same evaluation using the “large” evaluation set to see if this relationship was still present in a larger, more varied dataset. From Figures 8 and 9, it is apparent that the behaviour is very similar between both error rates.

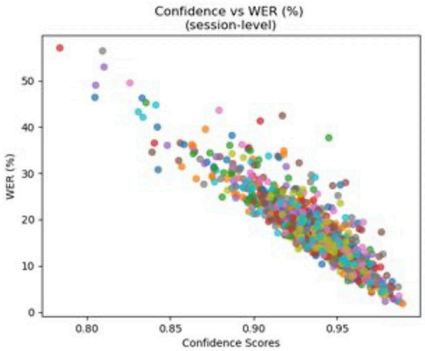


Figure 8: Aggregated confidence vs WER on the “large” evaluation set. Source: LSEG, as of November 22, 2024.

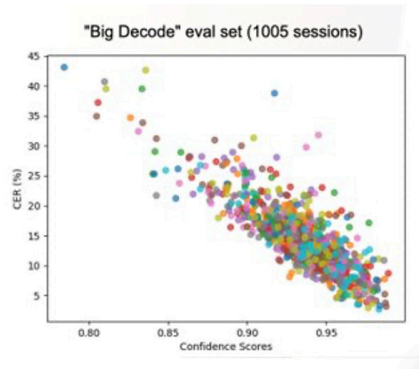


Figure 9: Aggregated confidence vs CER on the "large" evaluation set. Source: LSEG, as of November 22, 2024.

Between both datasets, and both error conditions, there is negative correlation between error rate and aggregated confidence.

2.3 Confidence Labels versus Confidence Scores

As illustrated in Figure 5, we decided to provide end users with a category label rather than a category score because, without calibration, confidence measures tend to result in numbers that are artificially high or overestimated. Calibration techniques can be used to mitigate this issue however we were reluctant to rely on these too heavily: these techniques tend to rely on model-specific learning, meaning calibration would have to be "relearnt" every time we updated our models. End-users who use this system every day may come to develop their own mental model of these statistics, only to have them disrupted. Instead, we wanted to abstract the numbers away from end-users and supply them instead with labels of "high", "medium", and "low" confidence. As such, the information would be presented in the manner outlined in Figure 10.

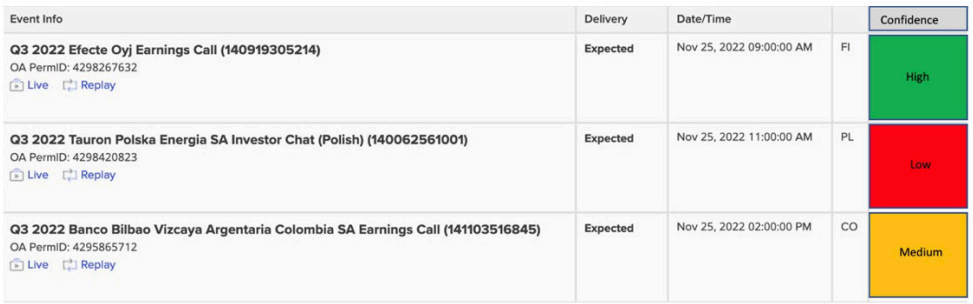


Figure 10: Wireframe demonstrating how we envisaged this information being presented to end-users in the UI. Source: LSEG, as of November 22, 2024.

With the presentation of such labels, the confidence scores represent a usefully additional metadata field that end-users can embed into their workflows.

2.4 Two-Factored Modelling Approach

As mentioned in the previous section, we decided to output confidence labels rather than scores due to potential complications caused by calibration. Another issue we had to overcome was the lack of labelled data. While we have access to a lot of ASR output with its accompanying confidence score and error rate, we do not have a gold-standard evaluation set with confidence labels provided by human annotators. Moreover, the decision of what constitutes a confidently recognised document is a subjective measure that could be impacted by editor experience and linguistic background. With this in mind, we decided to implement a two-factored modelling approach (Figure 11). This system would model error (CER/WER) explicitly using the evaluation data as outlined in Section 1.4.

Figure 11: Two-factored modelling approach where the output of Model A (predicted errors) is used as predictors for Model B. The final output is a predicted category label. Source: LSEG, as of November 22, 2024.

One benefit of a two-factored modelling approach is that we can make changes to one model without necessarily having to recalibrate the other. Additionally, this abstracts the produced confidence score away from the end user even further. This is key to preventing them from building their own internal assumptions of what a "confident" session looks like. Accents, audio quality, and topic could potentially cloud a user's judgement as to the success of the ASR system. When predicting an empirical metric (CER/WER), this is mitigated.

We decided to explore the use of linear regression for Model A and a set of heuristics for Model B, as outlined in the following sections.

Model A: Linear Regression Classifier

The "large" dataset described in section 1.4 was split into training and test sets. Using approximately 900 aggregated, document-level confidences and their equivalent error rates, a simple linear regression classifier was trained for both error metrics. These models were then used to predict the error rates for the remaining 100 or so data points in the held-out test set. The result of plotting the predicted relationship alongside the true error rates is shown in Figure 12.

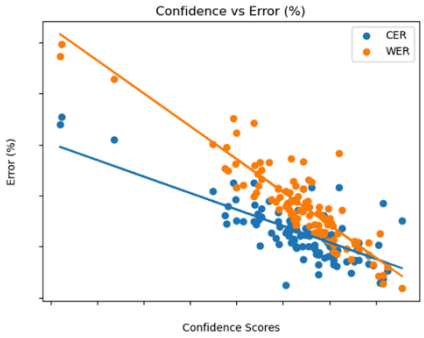


Figure 12: Line plot illustrating the correlation between error and document-level confidence score using the held-out test set. The true data points are plotted also. Source: LSEG, as of November 22, 2024.

For both WER and CER, the negative correlation between aggregated confidence and error has been captured by the model coefficients. Although there are no drastic outliers, there appears to be some variance from the modelled relationship between the two variables: this is something that will have to be considered when we come to model the label from our predicted error (as discussed in section 2.4.2). We evaluated the accuracy of these predictions using mean squared error (MSE) which is a typical metric to use for evaluating regression models, where the average squared difference between the observed and precited values are returned. The closer the MSE is to zero, the more accurate the model's predictions are. We returned the values outlined in the table below. From the results, we can see that the model trained to predict WER is slightly better, given the fact that the MSE is slightly lower. In the next section, we analyse the result of embedding these predictions into the two-factored modelling pipeline.

Error Metric	Mean Squared Error
WER	12.30
CER	14.07

Model B: Heuristics

The next question we had to answer was how to distinguish between “low”, “medium”, and “high” confidence sessions, given the error rates predicted in Section 2.4.1.

We initially used a k-means clustering algorithm for automatically learning the boundaries between confidence labels. After training the algorithm on the “large” dataset, we found the algorithm had split the data into the ranges outlined in the table below.

Confidence Label	Predicted Range
High	0.94 – 1.0
Medium	0.91 – 0.94
Low	0 – 0.91

Next, we decided to examine how these automatically learnt categories correlate with our interpretation of the error rates of high, medium, and low confidence documents, as illustrated in the figure below.

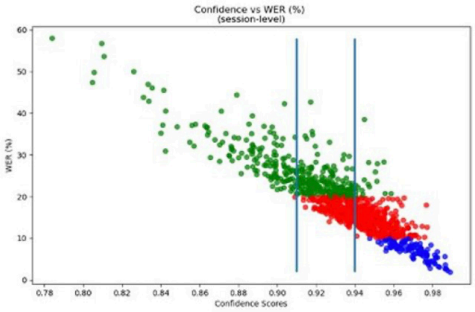


Figure 13: A comparison between k-means cluster-defined confidence boundaries (blue vertical lines) and our estimation of “high” (green), “medium” (red), and “low” (blue) scoring sessions. As illustrated, there is not a clear overlap between these variables. Source: LSEG, as of November 22, 2024.

Given the estimated error rates outlined in the table below, learned error boundaries do not correlate with the estimated boundaries.

Confidence Label	Estimated Error Rate
High	0 – 10%
Medium	10 – 25%
Low	25%+

With this in mind, we decided to forgo the use of the k-means clusters and instead use the estimated error rates from the table above as heuristics for predicting the confidence label. Although this is a simple solution, we found the performance to work very well, as illustrated in the next section.

2.5 Final Analysis on Validation Set

The Content Operations team provided a set of 20 files which constitute a new evaluation set. These files were selected by their team as being representative of the objective classifications that they gather as described below:

- ASR quality, scored on a scale of 1 to 5 with 5 being the best.
- Audio quality, scored on a scale of 1 to 5 with 5 being the best.

We also requested that this evaluation set be varied in terms of industry and region, as we tend to see some correlation between ASR performance and these metadata factors – in part, due to non-native English accents and the quality of audio equipment.

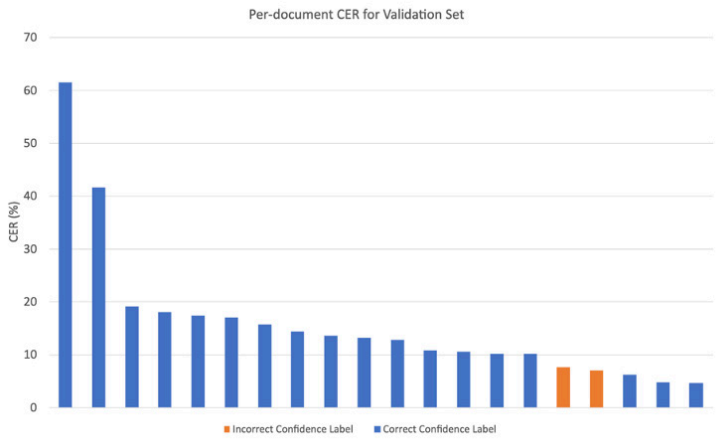


Figure 14: CER for each file in the validation set. Blue represents a session whose confidence label was correctly classified; orange represents a session whose label was incorrectly classified. From the graph, it is clear to see that there is error around the 6 – 9% CER threshold. Source: LSEG, as of November 22, 2024.

The sessions were processed by the recogniser and the CER and confidence labels were retrieved. Using the linear regression model trained on CER as outlined in section 2.4.1 and the heuristics outlined in section 2.4.2, we achieved a category label classification accuracy of 90% which shows that the model is successful in classifying the majority of categories. Clearly, there is some error around the 6-9% CER mark which suggests that the threshold for a high confidence session could be better estimated, perhaps by making use of other predictors, such as audio quality or number of speakers. We leave this open as a potential avenue for future work.

3. Discussion and Further Work

In this paper we have outlined a method for obtaining reliable document-level confidence scores from an ASR system. The technique outlined in this paper, although simple, appears to produce very encouraging results. Using a set of machine learning techniques and data analysis we find this solution to be robust as well as computationally efficient, however there are steps that could be taken to expand upon this work in the future.

As mentioned in Section 1.2, CESLT explored the use of confidence measures under both the factorised and end-to-end ASR architectures. All results in this paper report on the former system, however some experiments using an end-to-end system were also conducted. One challenge of using confidence scores from an end-to-end system such as Whisper[3] is the fact that deep neural networks tend to be particularly susceptible to the problem of overestimation [3] when using existing information from the decode. Further research is expected to be conducted in this area and may be the subject of a future white paper. Furthermore, the research in this paper focussed exclusively on the use of pre-existing information from the ASR decode: another possible avenue of future research could be the use of purpose-made, decode-agnostic features either on their own or in a hybrid approach. Finally, as we transition to more state-of-the art systems, the principles explored in this study continue to inform and guide ongoing advancements in our technologies.

The work described in this paper is now in production, and our downstream users have access to document-level confidence scores. Working in partnership with these users, we will continuously improve upon the efficiency and timesaving gains we intend to make with this project.

[3] Introducing Whisper | OpenAI

References

1. Mohri, Mehryar, and Michael Riley. "Weighted determinization and minimization for large vocabulary speech recognition." *Fifth European Conference on Speech Communication and Technology*. 1997.

2. Xu, Haihua, et al. "Minimum bayes risk decoding and system combination based on a recursion for edit distance." *Computer Speech & Language* 25.4 (2011): 802-828.

3. Guo, Chuan, et al. "On calibration of modern neural networks." *International conference on machine learning*. PMLR, 2017.