Self-supervised representation learning of spoken language

LOT School 2024

Grzegorz Chrupała

How is speech different from writing?

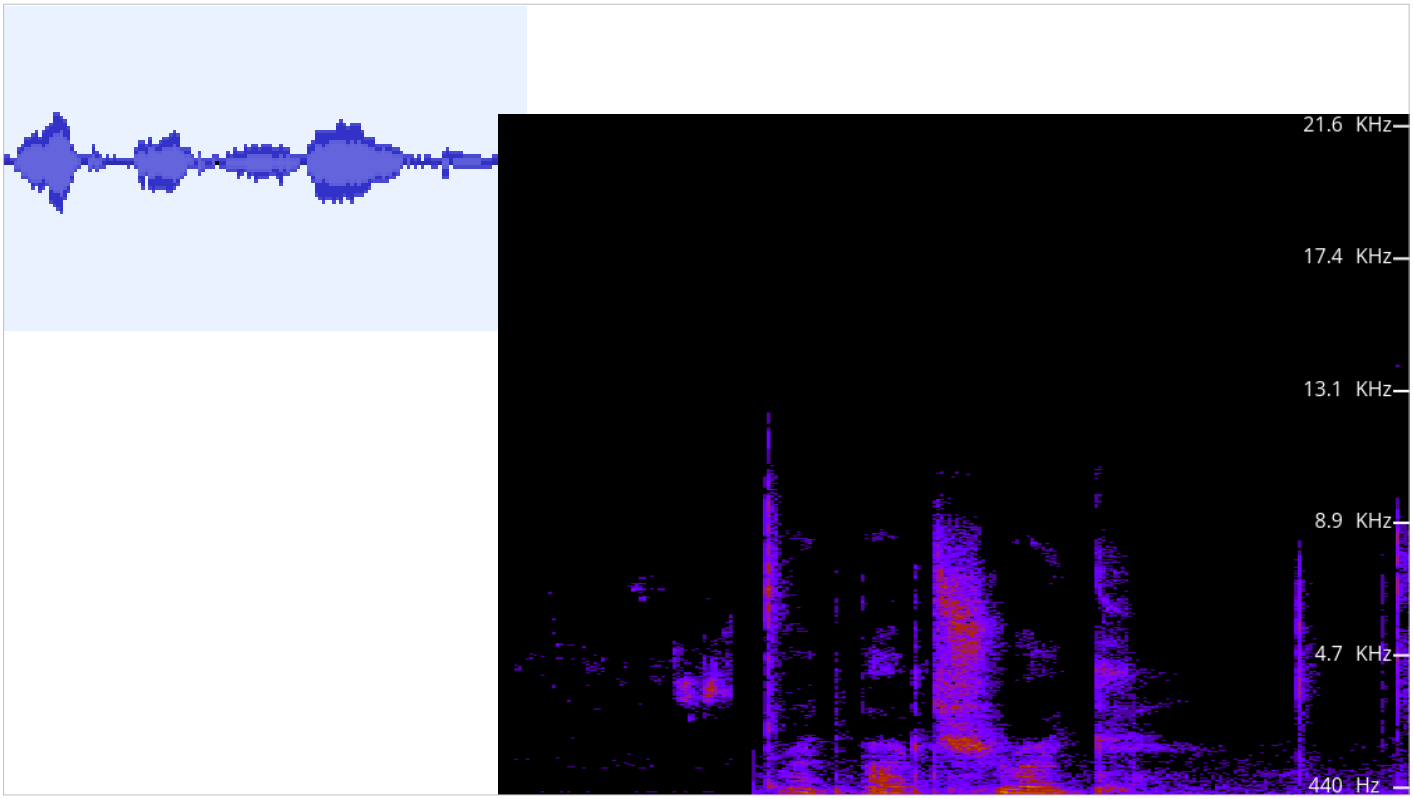
How can we represent speech in a computer?

Speech includes many features absent from writing: e.g. speaker features, prosody.

Crucially, a spoken utterance it is a continuous signal, while its transcriptions is a sequence of discrete symbols.

There are no breaks between words or phonemes in the audio signal.

When speech is recorded on a computer, this continuous signal is sampled, typically at 16kHz (telephone) or 44.1 kHz (Audio CD).



Speech and audio signal in general can be represented digitally:

- Waveform: signal amplitude sampled over time
- Spectrogram: spectrum of frequencies as it varies with time, plotted as time on the x-axis, frequency on the y-axis, and color representing intensity.
- MFCC: coefficients derived from the power spectrum of a signal, capturing its spectral characteristics in a form that mimics human auditory perception.

What are the challenges of a BERT-like or GPT-like language modeling approach for the spoken modality?

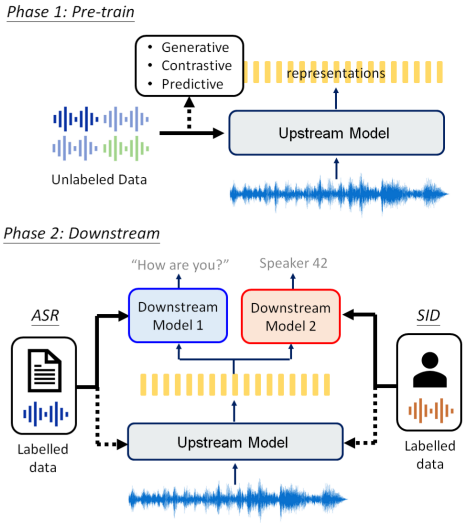
The sequences are much longer.

It is more challenging to come up with appropriate-sized units to predict or generate. Single frames correspond to very short segments of audio. Adjacent frames are very highly correlated.

Predicting discrete categories usually works better than regressing multidimensional vectors. But predicting discrete categories is not directly applicable here.

Self-Supervised Speech Representation Learning: A Review

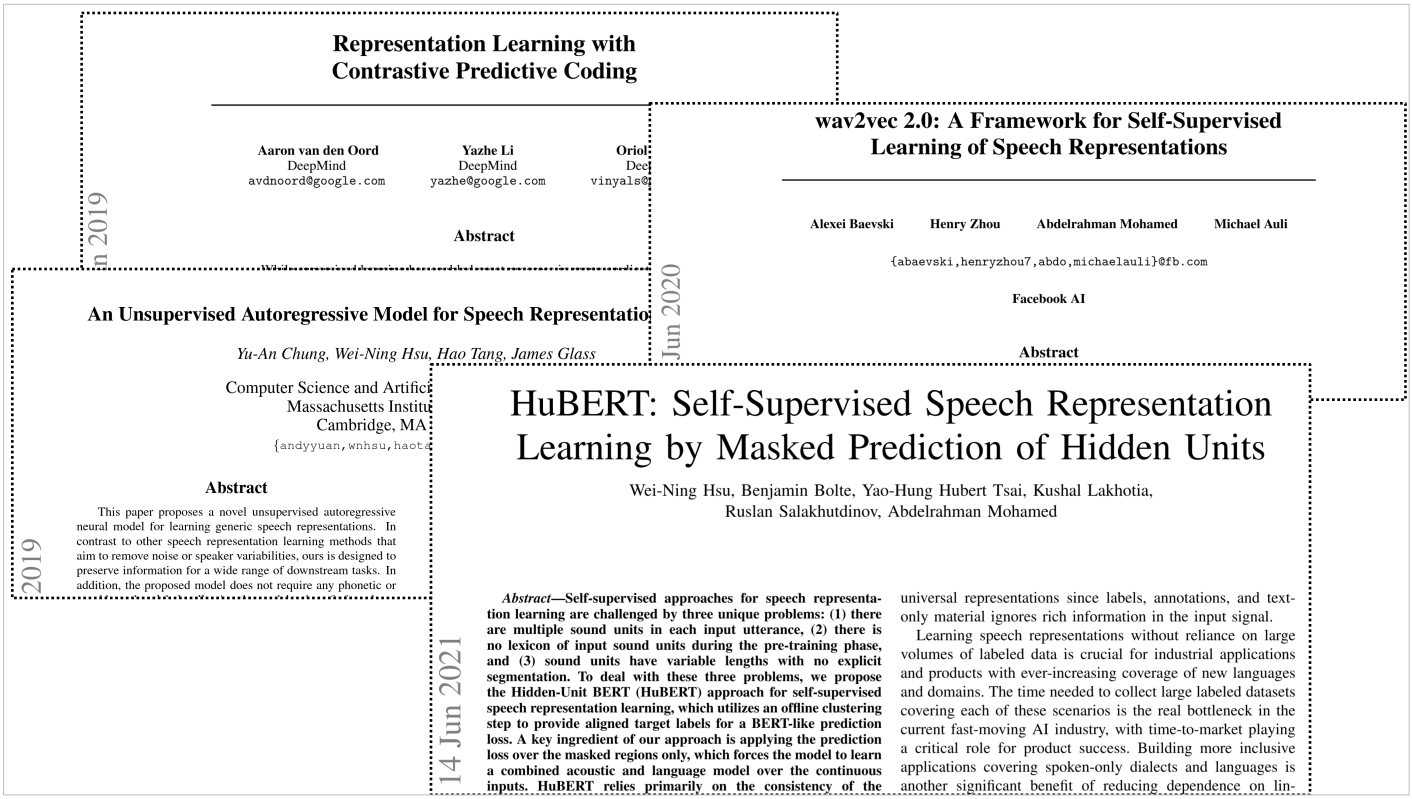
Abdelrahman Mohamed*, Hung-yi Lee*, Lasse Borgholt*, Jakob D. Havtorn*, Joakim Edin, Christian Igel
Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, Tara N. Sainath, Shinji Watanabe



Language processing been divided into **speech processing** on the one hand,
and
text-based NLP on the other hand.

Why is this the case?

Partly due to historical reasons, with speech processing requiring different digital signal-processing expertise, as opposed to text processing which use to be based on logic, discrete math, and later symbolic statistical modeling.

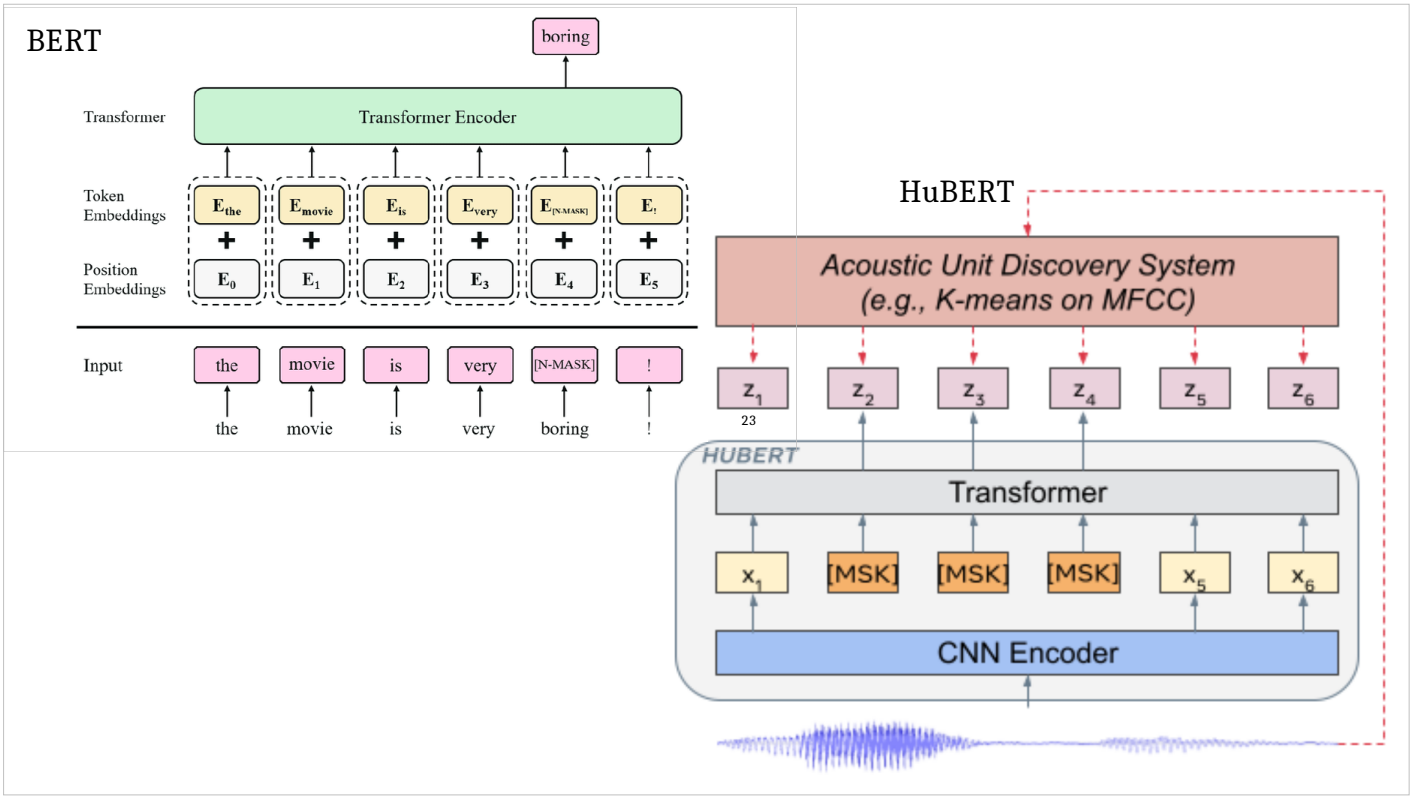


Spoken language modeling proved challenging.

In the last few years have been many different attempts to replicate the workings of textual language models with only partial success.

The first model to see widespread adoption was wav2vec2.

However, HuBERT is simpler to understand and closer to the original BERT formulation and is a good starting point to understand self-supervised spoken language modeling.

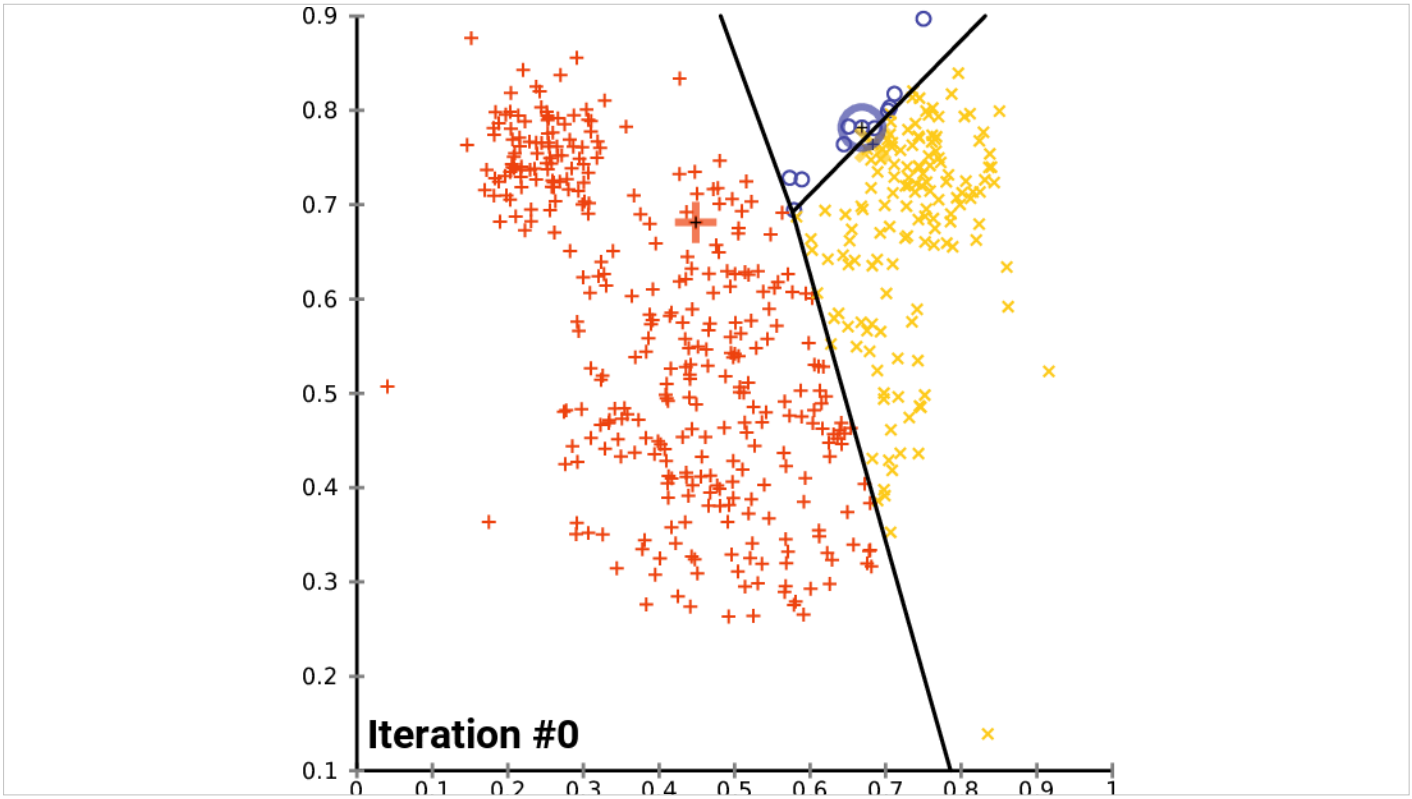


BERT image from
https://ankur3107.github.io/blogs/masked-langauge-modeling/
HuBERT image from
https://doi.org/10.1109/TASLP.2021.3122291

HuBERT is close to BERT. Specifically it is based on guessing the masked segment, encoded as a discrete symbol. The symbols are induced from the audio input via an off-line clustering module.

Initially the clustering is done on MFCC. In a second iteration the representation learned by the model is itself used as input to the clustering module.

While BERT uses word-embeddings as input, HuBERT features a CNN module which subsamples and encodes the waveform audio representation.

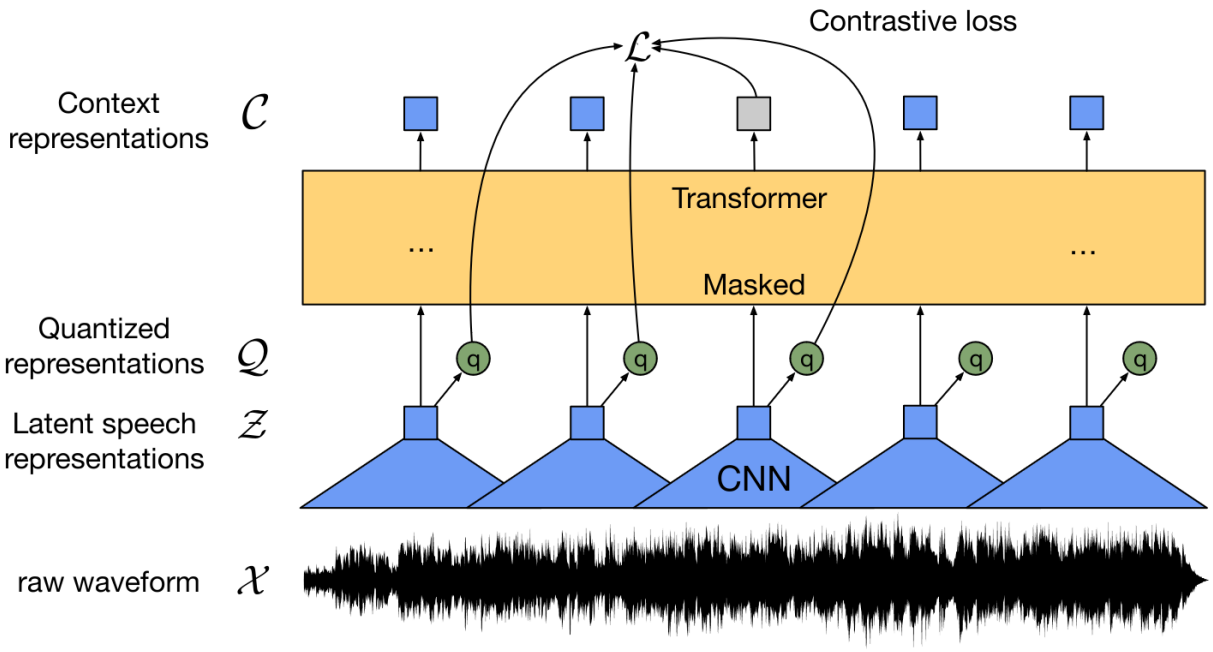


K-Means is a simple iterative clustering algorithm.
Visualization from
https://commons.wikimedia.org/wiki/File:K-means_convergence.gif

We alternate between two steps:
1) Assign data points to the cluster whose mean (centroid) is the closests;
2) Recompute centroids based on current assignment.

The algorithm can be initialized in various ways; in the simplest case the initial cluster means are random.

In HuBERT, the first run of K-means uses 100 clusters.
The second run (with learned representations as inputs) uses 500 clusters.



Many other self-supervised models of spoken language which predict or discriminate between discrete categories typically use a clustering step integrated within the overall architecture. This is also how wave2vec2 works.

Wav2vec2 has two main differences from HuBERT:

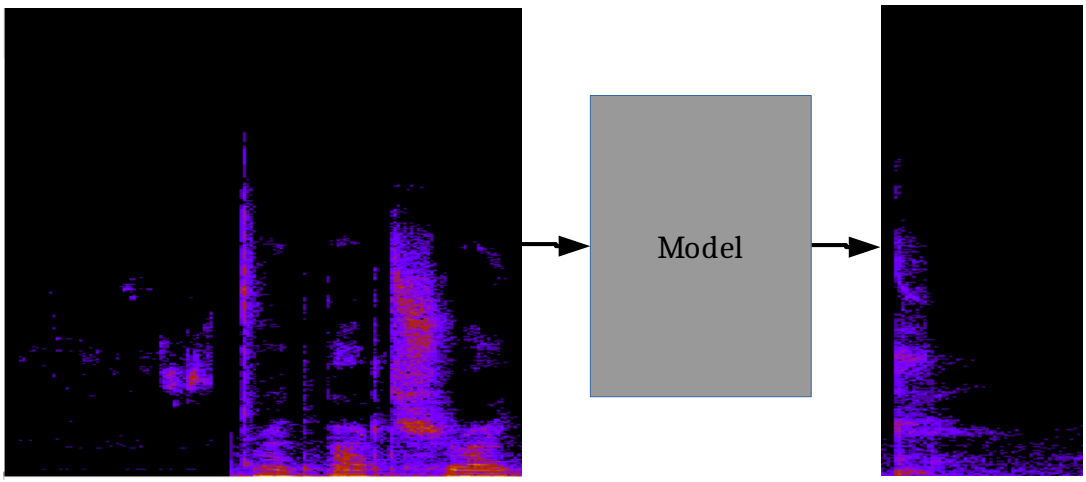
1) Clustering is done online, as part of the neural model;
2) Contrastive rather than predictive loss is used.

Image from https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html

Mohamed et al 2022 distinguish between: generative, contrastive & predictive approaches.

Which are the features of these categories?

Generative

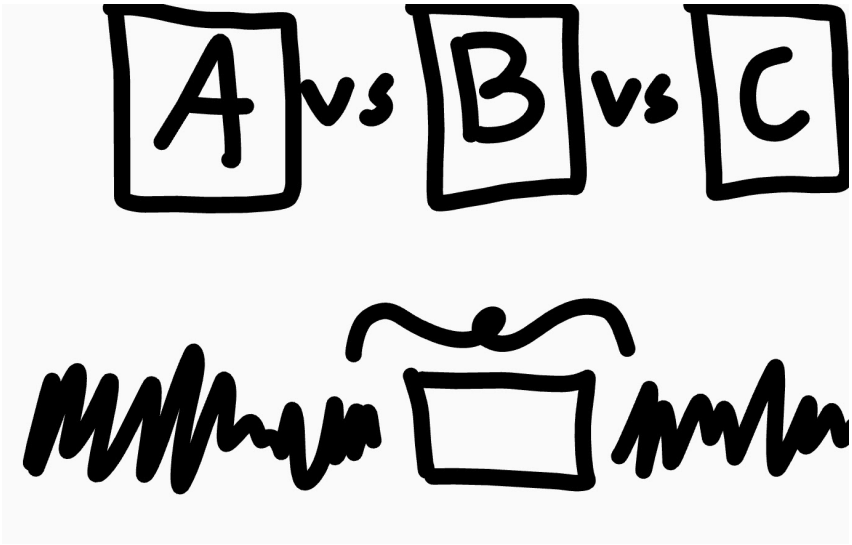


Not in the meaning of generating utterances – they mean that the (part) of the original input is the target of reconstruction.

For example, autoencoder-style approaches.

Or approaches known as autoregressive predictive coding, whose general idea is depicted here.

Contrastive



In a constrastive approach the masked part of the input is not directly reconstructed. Rather, the objective is to discriminate between the true representation of the masked input, and distractors which represent other unrelated segments.

Here we are discriminating between internal representations of input segments, not direct input features, and we use the contrastive setting to prevent the model from learning trivial representations. Which could happen if the goal was to predict representations rather than discriminate between alternatives.

Predictive

In this case the representation (discrete) comes from an independent (offline) module (such as K-means), so there is no need to use the contrastive setting. The goal is to directly predict the representation of the masked segment.

Advantages and drawbacks of using internal vs offline discretization of representations.

Discrete choices inside a neural model introduce non-differentiable computations, which make is more complicated to apply standard gradient-based learning.

But there are techniques to circumvent this difficulty, and the result is a more self-contained model with no need for iterating between independent disconnected modules.

What are the limitations of the self-supervised spoken language models?

From an engineering perspective?

And from a cognitive modeling perspective?

Image from
http://808lsalvador.deviantart.com/art/babies-like-listening-to-music-273884785

From a cognitive modeling perspective, it's equivalent to a baby learning a language by listening to the radio.

From an engineering perspective (and also cognitive) we may (possibly) get better sample efficiency, and better representations by exploiting language-externals sources of supervision.