Towards Detection and Remediation of Phonemic Confusion

Francois Roewer-Despres[1*] **Arnold YS Yeung**[1*] **Ilan Kogan**[2*]
[1]Department of Computer Science [2]Department of Statistics
University of Toronto
{francoisrd, arnoldyeung}@cs.toronto.edu
mail@ilankogan.ca

Abstract

Reducing communication breakdown is critical to success in interactive NLP applications, such as dialogue systems. To this end, we propose a confusion-mitigation framework for the detection and remediation of communication breakdown. In this work, as a first step towards implementing this framework, we focus on detecting phonemic sources of confusion. As a proof-of-concept, we evaluate two neural architectures in predicting the probability that a listener will misunderstand phonemes in an utterance. We show that both neural models outperform a weighted n-gram baseline, showing early promise for the broader framework.

1 Introduction

Ensuring that interactive NLP applications, such as dialogue systems, communicate clearly and effectively is critical to their long-term success and viability, especially in high-stakes domains, such as healthcare. Successful systems should thus seek to reduce communication breakdown. One aspect of successful communication is the degree to which each party understands the other. For example, properly diagnosing a patient may necessitate asking logically complex questions, but these questions should be phrased as clearly as possible to promote understanding and mitigate confusion.

To reduce confusion-related communication breakdown, we propose that generative NLP systems integrate a novel confusion-mitigation framework into their natural language generation (NLG) processes. In brief, this framework ensures that such systems avoid transmitting utterances with high predicted probabilities of confusion. In the simplest and most decoupled formulation, an existing NLG component simply produces alternatives to any rejected utterances without additional guiding information. In more advanced and coupled formulations, the NLG and confusion prediction components can be closely integrated to better determine precisely how to avoid confusion. This process can also be conditioned on models of the current listener or task to achieve personalized or context-dependent results. Figure 1 shows the simplest variant of the framework.

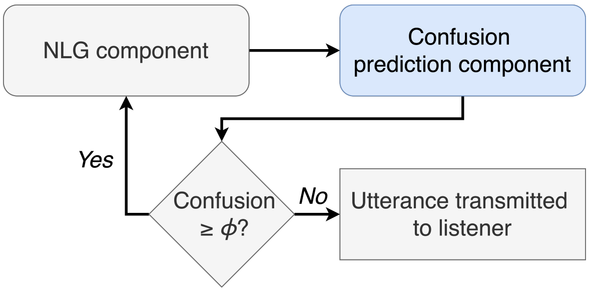

NLG component
Confusion prediction component
Yes
Confusion ≥ ϕ?
No
Utterance transmitted to listener

Figure 1: A simplified variant of our proposed confusion-mitigation framework, which enables generative NLP systems to detect and remediate confusion-related communication breakdown. The confusion prediction component predicts the confusion probability of candidate utterances, which are rejected if this probability is above a decision threshold, ϕ.

As a first step towards implementing this framework, we work towards developing its central confusion prediction component, which predicts the confusion probability of an utterance. In this work, we specifically target phonemic confusion, that is, the misidentification of heard phonemes by a listener. We consider two potential neural architectures for this purpose: a fixed-context, feed-forward network and a residual, bidirectional LSTM network. We train these models using a novel proxy data set derived from audiobook recordings, and compare their performance to that of a weighted n-gram baseline.

[*] Equal contribution.

2 Background and Related Work

Prior work focused on identifying confusion in natural language, rather than proactively altering it to help reduce communication breakdown, as our framework proposes. For example, Batliner et al. (2003) showed that certain features of recorded speech (e.g., repetition, hyperarticulation, strong emphasis) can be used to identify communication breakdown. The authors relied primarily on prosodic properties of recorded phrases, rather than the underlying phonemes, words, or semantics, for identifying communication breakdown. On the other hand, conversational repair is a turn-level process in which conversational partners first identify and then remediate communication breakdown as part of a trouble-source repair (TSR) sequence (Sacks et al., 1974). Using this approach, Orange et al. (1996) identified differences in TSR patterns amongst people with no, early-stage, and middle-stage Alzheimer's, highlighting the usefulness of communication breakdown detection. However, such work does not directly address the issue of proactive confusion mitigation and remediation, which the more advanced formulation of our framework aims to address through listener and task conditioning. Our focus is on the simpler formulation in this preliminary work.

Rothwell (2010) identified four types of noise that may cause confusion: physical noise (e.g., a loud highway), physiological noise (e.g., hearing impairment), psychological noise (e.g., attentiveness of listener), and semantic noise (e.g., word choice). We postulate that mitigating confusion resulting from each type of noise may be possible, at least to some extent, given sufficient context to make an informed compensatory decision. For example, given a particularly physically noisy environment, speaking loudly would seem appropriate. Unfortunately, such contextual information is often lacking from existing data sets. In particular, the physiological and psychological states of listeners is rarely recorded. Even when such information is recorded (e.g., in Alzheimer's speech studies Orange et al., 1996), the information is very coarse (e.g, broad Alzheimer's categories such as `none`, `early-stage`, and `middle-stage`).

We leave these non-trivial data gathering challenges as future work, instead focusing on phonemic confusion, which is significantly easier to operationalize. In practice, confusion at the phoneme-level may arise from any category of Rothwell noise. It may also arise from the natural similarities between phonemes (discussed next). While many of these will not be represented in the text-based phonemic transcriptions data set used in this preliminary work, our approach can be extended to include them.

Researchers in speech processing have studied the prediction of phonemic confusion but, to our knowledge, this work has not been adapted to utterance generation. Instead, tasks such as preventing of sound-alike medication errors (i.e., naming medications so that two medications do not sound identical) are common (Lambert, 1997). Zgank and Kacic (2012) showed that the potential confusability of a word can be estimated by calculating the Levenshtein distance (Levenshtein, 1966) of its phonemic transcription to that of all others in the vocabulary. We take inspiration from Zgank and Kacic (2012) and employ a phoneme-level Levenshtein distance approach in this work.

In the basic definition of the Levenshtein distance, all errors are equally weighted. In practice, however, words that share many similar or identical phonemes are more likely to be confused for one another. Given this, Sabourin and Fabiani (2000) developed a *weighted* phoneme-level Levenshtein distance, where weights are determined by a human expert or a learned model, such as a hidden Markov model. Unfortunately, while these weights are meant to represent phonemic similarity, selecting an appropriate distance metric in phoneme space is non-trivial. The classical results of Miller (1954) and Miller and Nicely (1955) group phonemes experimentally based on the noise level at which they become indiscernible. The authors identify voicing, nasality, affrication, duration, and place of articulation as sub-phoneme features that predict a phoneme's sensitivity to distortion, and therefore measure its proximity to others. Unfortunately, later work showed that these controlled conditions do not map cleanly to the real world (Batliner et al., 2003). In addition, Wickelgren (1965) found alternative phonemic distance features that could be adapted into a distance metric.

While this prior research sought to directly define a distance metric between phonemes based on sub-phoneme features, since no method has emerged as clearly superior, researchers now favour direct, empirical measures of confusability (Bailey and Hahn, 2005). Likewise, our work assumes that these classical feature-engineering approaches to predicting phoneme confusability can be improved upon with neural approaches, just as automatic speech recognition (ASR) systems have been improved through the use of similar methods (e.g., Seide et al., 2011; Zeyer et al., 2019; Kumar et al., 2020). In addition, these classical approaches do not account for context (i.e., other phonemes surrounding the phoneme of interest), whereas our approach conditions on such context to refine the confusion estimate.

3 Data

3.1 Data Gathering Process

To predict the phonemic confusability of utterances, we would ideally use a data set in which each utterance is annotated with speaker phonemic transcription (the reference transcription), as well as listener perceived phonemic transcription (the hypothesis transcription). We could then compare these transcriptions to identify phonemic confusion.

To the best of our knowledge, a data set of this type does not exist. The English Consistent Confusion Corpus contains a collection of individual words spoken against a noisy background, with human listener transcriptions (Marxer et al., 2016). This is similar to our ideal data set, however the words are spoken in isolation, and thus without any utterance context. This same issue arises in the Diagnostic Rhyme Test and its derivative data sets (Voiers et al., 1975; Greenspan et al., 1998). Other corpora, such as the BioScope Corpus (Vincze et al., 2008) and the AMI Corpus (Carletta et al., 2005), contain annotations of dialogue acts, which represent the intention of the speaker in producing each utterance (e.g., asking a question is labeled with the dialogue act `elicit_information`). However, dialogue acts relating to confusion only appear when a listener explicitly requests clarification from the speaker. This does not provide fine-grained information regarding which phonemes caused the confusion, nor does it capture any instances of confusion in which the listener does not explicitly vocalize their confusion.

We thus create a new data set for this work (Figure 2). The Parallel Audiobook Corpus contains 121 hours of recorded speech data across 59 speakers (Ribeiro, 2018). We use four of its audiobooks: *Adventures of Huckleberry Finn*, *Emma*, *Treasure Island*, and *The Adventures of Sherlock Holmes*. Crucially, the audio recordings in this corpus are aligned with the text being read, which allows us to create aligned reference and hypothesis transcriptions. For each text-audio pair, the text simply becomes the reference transcriptions, while a transcriber converts the audio into hypothesis transcriptions. Given the preliminary nature of this work, we create a proxy data set in which we use Google Cloud's publicly-available ASR system as a proxy for human transcribers (Cloud, 2019). We then process these transcriptions to identify phonemic confusion events (as described in Section 3.2). The final data set contains 84,253 parallel transcriptions. We split these into 63,189 training, 10,532 validation, and 10,532 test transcriptions (a 75%-12.5%-12.5% split). The average reference and hypothesis transcription lengths are 65.2 and 62.3 phonemes, respectively. The transcription error rate (i.e., the proportion of phonemes that are mis-transcribed) is only 8%, so there is significant imbalance in the data set.

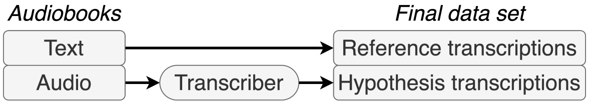


Figure 2: We create a new data set with parallel reference and hypothesis transcriptions from audiobook data with parallel text and audio recordings. The text simply becomes the reference transcriptions. A transcriber converts the audio recordings into hypothesis transcriptions. In this preliminary work, we use an ASR system as a proxy for human transcribers.

For the purposes of this preliminary work, the Google Cloud ASR system (Cloud, 2019) is an acceptable proxy for human transcription ability under the reasonable assumption that, for any particular transcriber, the distribution of error rates across different phoneme sequences is nonuniform (i.e., within-transcriber variation is present). This assumption holds in all practical cases, and is reasonable since the confusion-mitigation framework we propose can be conditioned on different transcribers to control for inter-transcriber variation as future work.

3.2 Transcription Error Labeling

We post-process our aligned reference-hypothesis transcription data set in two steps. First, each transcription must be converted from the word-level to the phoneme-level. For this, we use the CMU Pronouncing Dictionary (Weide, 1998), which is based on the ARPAbet symbol set. For any words with multiple phonemic conversions, we simply

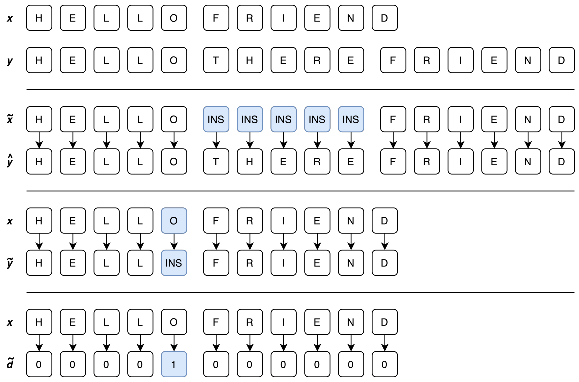

x H E L L O F R I E N D
y H E L L O T H E R E F R I E N D
x̃ H E L L O INS INS INS INS INS F R I E N D
ŷ H E L L O T H E R E F R I E N D
x H E L L O F R I E N D
ỹ H E L L INS F R I E N D
x H E L L O F R I E N D
d̃ 0 0 0 0 1 0 0 0 0 0 0

Figure 3: Illustration of our transcription error labeling process (using letters instead of phonemes for readability). Given aligned reference ($\boldsymbol{x}$) and hypothesis ($\boldsymbol{y}$) vectors, we use the Levenshtein algorithm to ensure they have the same length. Because $\boldsymbol{y}$ is not available at test time, we then "collapse" consecutive insertion tokens to force the vectors to have the original length of $\boldsymbol{x}$. Finally, we replace $\tilde{\boldsymbol{y}}$ with the binary vector $\tilde{\boldsymbol{d}}$, which has 1's wherever $\boldsymbol{x}$ and $\tilde{\boldsymbol{y}}$ don't match.

default to the first conversion returned by the API.

Second, we label each resulting phoneme in each reference transcription as either correctly or incorrectly transcribed. This is nontrivial, because the number of phonemes in the reference and hypothesis transcriptions are rarely equal, and thus require phoneme-level alignment. For this purpose, we use a variant of the phoneme-level Levenshtein distance that returns the actual alignment, rather than the final distance score (Figure 3).

Formally, let $\boldsymbol{x} \in \mathbb{K}^a$ be a vector of reference phonemes and $\boldsymbol{y} \in \mathbb{K}^b$ be a vector of hypothesis phonemes from the data set. $\mathbb{K}$ refers to the set $\{1, 2, 3, \ldots, k, \texttt{<INS>}, \texttt{<DEL>}, \texttt{<SOS>}, \texttt{<EOS>}\}$, where k is the number of unique phonemes in the language being considered (e.g., in English, $k \approx 40$ depending on the dialect). In general, $a \neq b$, but we can manipulate the vectors by incorporating insertion, deletion, and substitution tokens (as done in the Levenshtein distance algorithm). In general, this yields two vectors of the same length, $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}} \in \mathbb{K}^c, c = \max(a, b)$. While this manipulation can be performed at training time because $\boldsymbol{y}$ and b are known, such information is unavailable at test time. Therefore, we modify the alignment at training time to ensure $\tilde{\boldsymbol{x}} \equiv \boldsymbol{x}$ and $c \equiv a$. To achieve this, we "collapse" consecutive insertion tokens into a single instance of the insertion token, which ensures that $|\tilde{\boldsymbol{y}}| = a$.

Additionally, we assume that each hypothesis phoneme, $\tilde{y}_i \in \tilde{\boldsymbol{y}}$, is conditionally independent of the others. That is, $P(\tilde{y}_i = x_i \,|\, \boldsymbol{x}, \tilde{y}_{\neq i}) = P(\tilde{y}_i = x_i \,|\, \boldsymbol{x})$.[1] We hypothesize that this assumption, similar to the conditional independence assumption of Naïve Bayes (Zhang, 2004), will still yield directionally-correct results, while drastically increasing the tractability of the computation.

This assumption also allows us to simplify the output space of the problem. Specifically, since we only care to predict $P(\tilde{\boldsymbol{y}} \neq \boldsymbol{x})$, with this assumption, we now only need to consider, for each i, whether $\tilde{y}_i = x_i$, rather than dealing with the much harder problem of predicting the exact value of $\tilde{y}_i$. To achieve this, we use an element-wise Kronecker delta function to replace $\tilde{\boldsymbol{y}}$ with a binary vector, $\tilde{\boldsymbol{d}}$, such that $\tilde{d}_i \leftarrow \tilde{y}_i \neq x_i$. Thus, the binary vector $\tilde{\boldsymbol{d}}$ records the position of each transcription error, that is, the position of each phoneme in $\boldsymbol{x}$ that was confused.

With the $\boldsymbol{x}$'s as inputs and the $\tilde{\boldsymbol{d}}$'s as ground truth labels, we can train models to predict $P(\tilde{d}_i \,|\, \boldsymbol{x})$ for each i. As a post-processing step, we can then combine these individual probabilities to estimate the utterance-level probability of phonemic confusion, $P(\tilde{\boldsymbol{y}} \neq \boldsymbol{x})$, which is the output of the central confusion prediction component in Figure 1.

This formulation is general in the sense that any x_i can affect the predicted probability of any $\tilde{d}_i$. In practice, however, and especially for long utterances, this is overly conservative, as only nearby phonemes are likely to have a significant effect. In Section 4, we describe any additional conditional independence assumptions that each architecture makes to further simplify its probability estimate.

4 Model Architectures and Baseline

With recent advances, various neural architectures have been applied to NLP tasks. Early work includes n-gram-based, fully-connected architectures for language modeling tasks (Bengio et al., 2003; Mikolov et al., 2013). Recurrent neural network (RNN) architectures were then shown to be successful for applications such as language modeling, speech recognition, and phoneme recognition (Graves and Schmidhuber, 2005; Mikolov et al., 2011). RNN architectures such as the LSTM (Hochreiter and Schmidhuber, 1997) and GRU (Chung et al., 2015) variants had been successful in many NLP applications, such as machine language translation and phoneme classification (Sundermeyer et al., 2012; Graves et al., 2013; Graves

[1] $\tilde{y}_{\neq i}$ is every element in $\tilde{\boldsymbol{y}}$ except the one at position i.

and Schmidhuber, 2005). Recently, the transformer architecture (Vaswani et al., 2017), which uses attention instead of recurrence to form dependencies between inputs, has shown state-of-the-art results in many areas of NLP, including syllable-based tasks (e.g., Zhou et al., 2018).

In this work, we propose a fixed-context-window architecture and a residual bi-LSTM architecture for the central component of our confusion-mitigation framework. While similar architectures have already been applied to phoneme-based applications, such as phoneme recognition and classification (Graves and Schmidhuber, 2005; Weninger et al., 2015; Graves et al., 2013; Li et al., 2017), to our knowledge, our study is the first to apply these architectures to identify phonemes related to confusion for listeners. In our opinion, these architectures strike an acceptable balance between compute and capability for this current work, unlike the more advanced transformer architectures, which require significantly more resources to train.[2]

Since the data set is imbalanced (see Section 3.1), without sample weighting, early experiments showed that both architectures never identified any phonemes as likely to be mis-transcribed (i.e., high specificity, low sensitivity). Accordingly, since the imbalance ratio is approximately 1:10, transcription errors are given 10-times more weight than properly-transcribed phonemes in our binary cross-entropy loss function.

4.1 Fixed-Context Network

The fixed-context network takes as input the current phoneme, x_i, and the 4 phonemes before and after it as a fixed window of context (Figure 4a). This results in the additional conditional independence assumption that $P(\tilde{d}_i \,|\, \boldsymbol{x}) = P(\tilde{d}_i \,|\, x_{i-4:i+4})$. That is, only phonemes within the fixed context window of size 4 can affect the predicted probability of $\tilde{d}_i$.

These 9 phonemes are first embedded in a 15-dimensional embedding space. The embedding layer is followed by a sequence of seven fully-connected hidden layers with 512, 256, 256, 128, 128, 64, and 64 neurons respectively. Each layer is separated by Rectified Linear Unit (ReLU) non-linearities (Nair and Hinton, 2010; He et al., 2016). Finally, an output with a sigmoid activation function predicts the probability of a transcription error. We train with minibatches of size 32, using the Adam optimizer with parameters $\alpha = 0.001$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$ (Kingma and Ba, 2014) to optimize a 1:10 weighted binary cross-entropy loss function. We explored alternative parameter settings, and in particular a larger number of neurons, but found this architecture to be the most stable and highest performing of all variants tested, given the nature and relatively small size of the data set.

4.2 LSTM Network

The LSTM network receives the entire reference transcription, $\boldsymbol{x}$, as input and predicts the entire binarized hypothesis transcription, $\tilde{\boldsymbol{d}}$, as output (Figure 4b). Since the LSTM is bidirectional, we do not introduce any additional conditional independence assumptions. Each input phoneme is passed through an embedding layer of dimension 42 (equal to $|\mathbb{K}|$) followed by a bidirectional LSTM layer and two residual linear blocks with ReLU activations (He et al., 2016). An output residual linear block with a sigmoid activation predicts the probability of a transcription error. These skip connections are added since residual layers tend to outperform simpler alternatives (He et al., 2016). Passing the embedded input via skip connections ensures that the original input is accessible at all depths of the network, and also helps mitigate against any vanishing gradients that may arise in the LSTM.

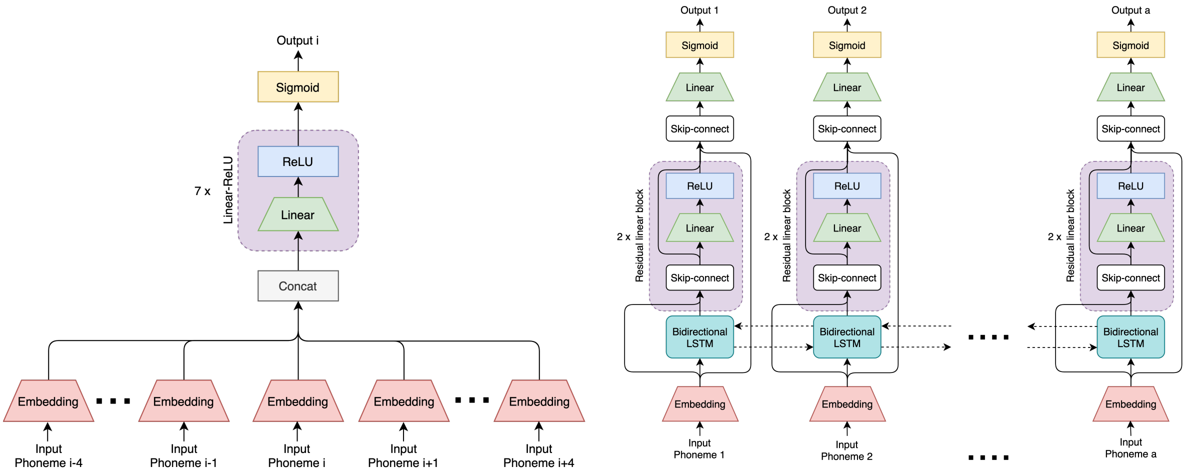
We use the following output dimensions for each layer: 50 for LSTM hidden and cell states, 40 for the first residual linear block, and 10 for the second. We train with minibatches of size 256, using the Adam optimizer with parameters $\alpha = 0.00005$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$ (Kingma and Ba, 2014) to optimize a 1:10 weighted binary cross-entropy loss function.

4.3 Weighted n-Gram Baseline

We compare our neural models to a weighted n-gram baseline model. That is, $\tilde{d}_i$ depends only on the n previous phonemes in $\boldsymbol{x}$ (an order-n Markov assumption). Formally, we make the conditional independence assumption that $P(\tilde{d}_i \,|\, \boldsymbol{x}) = p(\tilde{d}_i \,|\, x_{i-n+1:i})$. Extending this baseline model to include future phonemes would violate the order-n Markov assumption that is standard in n-gram approaches. In this preliminary work, we opt to keep the baseline as standard as possible.

A weighted n-gram model is computed using an algorithm similar to the standard maximum likelihood estimation (MLE) n-gram counting algo-

[2] Link to code: https://github.com/francois-rd/phonemic-confusion



(a) The fixed-context network uses a fixed window of context of size 4. These 9 phonemes are embedded using a shared embedding layer, concatenated, and then passed through 7 linear layers with ReLU activations, followed by an output layer with a sigmoid activation.

(b) Unrolled architecture of the LSTM network. The architecture consists of one bidirectional LSTM layer, two residual linear blocks with ReLU activations, and an output residual linear block with a sigmoid activation. Additional skip connections are added throughout.

Figure 4: Architectural variants of the confusion prediction component of our confusion-mitigation framework.

rithm, but with the introduction of a weighting scheme to deal with the class imbalance issue. The weighting is necessary for a fair comparison to the weighted loss function used in the neural network models. This approaches generalizes the standard MLE n-gram counting algorithm, which implicitly uses a weight of 1.

Formally, let $W > 0$ be the selected weight, and define $c_i \equiv x_{i-n+1:i}$ to simplify the notation. Also, let $C(\tilde{d}_i \,|\, c_i)$ be the *count* of all incorrect phoneme transcriptions in the context c_i in the entire data set, and similarly, $C(1 - \tilde{d}_i \,|\, c_i)$ for correct transcriptions.[3] The weighted n-gram is then computed as follows:

$$P(\tilde{d}_i \,|\, c_i) = \frac{W \times C(\tilde{d}_i \,|\, c_i)}{C(1 - \tilde{d}_i \,|\, c_i) + W \times C(\tilde{d}_i \,|\, c_i)}$$

Empirically, we find that a weighted 3-gram model works best; larger contexts are too sparse given the size of the data set and smaller contexts lack expressive capacity. We do not use any n-gram smoothing methods. Instead, any missing contexts encountered at test time are simply marked as incorrect predictions. For this particular data set, such missing contexts are vanishingly rare (occurring only 0.003% of the time), which justifies our approach.

5 Results and Discussion

5.1 Quantitative Analysis

We report receiver operating characteristic (ROC) curves for all models (Figure 5). To facilitate fair comparison, all models are trained with the same random ordering of training data in each epoch. Both neural network architectures outperform the weighted n-gram baseline by a small margin, with the fixed-context network appearing to perform slightly better overall. While no individual model exhibits any significant performance gain over the others, all models perform significantly better than random chance. This shows the promise of our framework, which is precisely the objective of this work. We next speculate as to the causes of the slight gaps that are observed.

The neural network models likely outperform the weighted n-gram baseline for multiple reasons. First, both neural network models condition on a context that includes both past and future phonemes (i.e., bidirectional), whereas the baseline only conditions on past phonemes (i.e., unidirectional). Utilizing future phonemes as context is useful since both humans and most state-of-the-art ASR systems use this information to revise their predictions. Second, the neural networks can learn subcontextual patterns that the baseline cannot. For example, the contexts `A B C` and `A B D` have the sub-context `A B` in common. Whereas the weighted n-gram treats these as completely dif-

[3] We slightly abuse the notation here. Recall that $\tilde{d}_i \leftarrow \tilde{y}_i \neq x_i$, so we notate $\tilde{y}_i = x_i$ as $1 - \tilde{d}_i$.

Ground Truth Phrase	Transcription of Audio Recording
... for they say **every body** is in love once ...	... for they say **everybody** is in love once ...
... his grave **looks shewed** that she was not ...	... his grave **look showed** that she was not ...
... shall use the carriage **to night** ...	... shall use the carriage **tonight** ...
... making him understand I **warn't** dead ...	... making him understand I **warrant** dead ...
... **shore** at that place so we **warn't** afraid ...	... **sure** at that place so we **weren't** afraid ...
... read Elton's **letter as** I was **shewn** in ...	... read Elton's **letters** I was **shown** in ...
... sacrifice my poor **hair to night** and ...	... sacrifice my poor **head tonight** and ...
... we **warn't** feeling just right ...	... we **weren't** feeling just right ...
... that there was no **want of taste** ...	... that there was no **on toothpaste** ...
... knew **that Arthur had** discovered ...	... knew **was it also have** discovered ...

Table 1: Randomly selected phrases from amongst the top 100 phonemes predicted to be *incorrectly* transcribed by the fixed-context model (transcription error probability > 0.999). Bold text denotes ASR transcription errors.

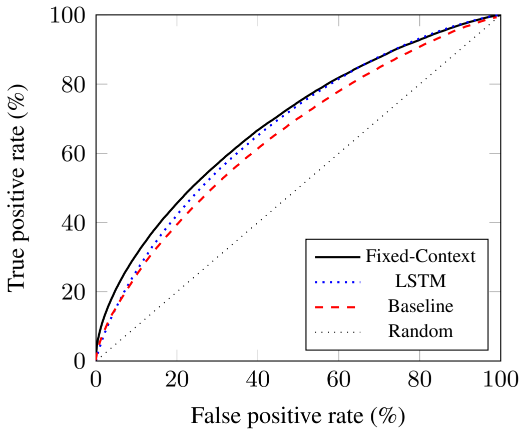


Figure 5: ROC curves for our model variants.

ferent contexts, the neural networks may be able to exploit the similarity between them. This kind of parameter sharing is more data efficient, which can lead to lower variance estimates (less overfitting) in the small data set setting we are considering.

The simpler fixed-context network slightly outperforms the more complex LSTM alternative. While RNN architectures have been shown to outperform feed-forward networks in language processing tasks (Sundermeyer et al., 2012), other research has shown that simpler architectures are still able to process phonemic data effectively (Ba and Caruana, 2014). The lack of an additional conditional independence assumption for the LSTM model may have resulted in worse data efficiency, since the model needs to expend parameters on all reference phonemes, even those very far away that may have little impact on the current one. In addition, the smaller number of parameters to estimate may have lead to lower variance in the fixed-context model. Given this, our avoidance of more advanced or deeper model, such as transformers, seems justified for this preliminary work. We hypothesize that such models could outperform all the models considered here given a significantly larger data set.

5.2 Qualitative Analysis

5.2.1 Description

We perform qualitative error analysis on randomly selected phonemes from amongst those that are most (Table 1) and least (Table 2) likely to contain transcription errors according to the fixed-context model. This offers some qualitative insights regarding phonemic confusion. We sample from the fixed-context model due to its slightly superior performance, and show small phrases centered around the phoneme most (least) likely to cause confusion, rather than full transcriptions, for clarity.

In addition, to improve readability, we show words rather than the underlying phonemes. As a result, some of the errors appear to be orthographic in nature even if they are not. For example, “every body” becomes “everybody” in the first example of Table 1. However, the phonemes that constitute “every body” and “everybody” are indeed different: “EH V **ER** IY B AA D IY” versus “EH V **R** IY B AA D IY”. As per our definitions in Section 3.2, these cases do represent transcription errors. However, it may be argued that such errors introduce unwanted noise in the data set, which we hope to correct in future work.

5.2.2 Analysis

First, we note that every sample in Table 1 does indeed have a transcription error, while few sam-

Ground Truth Phrase	Transcription of Audio Recording
... the exquisite feelings of delight and ...	... the exquisite feelings of delight and ...
... gone Mister Knightley called ...	... gone Mister Knightley called ...
... has been exceptionally ...	... has been exceptionally ...
... not afraid **of your** seeing ...	... not afraid **if you're** saying ...
... the sale of Randalls was long ...	... the sale of Randalls was long ...
... her very kind reception **of** himself ...	... her very kind reception **to** himself ...
... for the purpose of preparatory inspection ...	... for the purpose of preparatory inspection ...
... you would not be happy until you ...	... you would not be happy until you ...
... with the exception of this little blot ...	... with the exception of this little blot ...
... night we were in a great bustle getting ...	... night we were in a great bustle getting ...

Table 2: Randomly selected phrases from amongst the top 100 phonemes predicted to be *correctly* transcribed by the fixed-context model (transcription error probability < 0.03). Bold text denotes ASR transcription errors.

ples have errors in Table 2. It therefore seems as though, when the fixed-context model is very certain about the presence or absence of errors, it is usually correct.

Second, many of the transcription errors in Table 1 are seemingly caused by the archaic or idiosyncratic writing present in the books used to create the data set. While this can be seen as a source of unwanted noise (we used an ASR system trained on standard modern English), we argue that, as per Rothwell's model of communication (Section 2), familiarity with the vocabulary is, in fact, a very legitimate source of semantic noise. Indeed, phrases using more modern and standard vernacular are seemingly less likely to be confusing, according to the fixed-context model.

Third, many of the errors not related to archaism involve stop words, homonyms, or near-homophones, which intuitively makes sense. Additionally, hard consonant sounds between words (and stress at the beginning rather than at the end of words) appears more common in the set of correctly-transcribed phrases as compared to the set of incorrectly-transcribed ones. These findings suggest the fixed-context model has picked up on some underlying patterns governing phonemic confusion, which is promising for our confusion-mitigation framework as a whole.

5.3 Future Work

This work uses a relatively small data set. Creating and using a significantly larger corpus using human subjects rather than an ASR proxy would likely yield more directly relevant results. We postulate that, with a larger and higher quality data set, a deeper and more advanced neural network architecture, such as the transformer, may produce stronger results. Future work can also investigate the differences in human phonemic confusability on 'natural' versus semantically-unpredictable sentences.

A major aspect of our confusion-mitigation framework, which we have not explored in this work, is the generation of alternative, clearer utterances that retain the initial meaning. Constructively enumerating these alternatives is non-trivial, as is identifying the neighbourhood beyond which their meaning differs too significantly from the original. Conditioning on a specific listener's priors as an additional mechanism to reduce communication breakdown is another major aspect we leave to future work.

Perhaps most significantly, we have limited the scope of our confusion assessment drastically in this preliminary work, primarily to simplify the data gathering process. While our results are promising, communication breakdown is a nuanced and multi-faceted phenomenon of which phonemic confusion is but one small component. Modeling these larger and more complex processes remains an important open challenge.

6 Conclusion

Reducing communication breakdown is critical to successful interaction in dialogue systems and other generative NLP systems. In this work, we proposed a novel confusion-mitigation framework that such systems could employ to help minimize the probability of human confusion during an interaction. As a first step towards implementing this framework, we evaluated two potential neural architectures—a fixed-context network and an LSTM network—for its central component, which predicts the confusion probability of a candidate utterance. These neural architectures outperformed a weighted n-gram baseline (with the fixed-context network performing best overall) when trained using a proxy data set derived from audiobook recordings. In addition, qualitative analyses suggest that the fixed-context model has uncovered some of the more intuitive causes of phonemic confusion, including stop words, homonyms, near-homophones, and familiarity with the vocabulary. These preliminary results show the promise of our confusion-mitigation framework. Given this early success, further investigation and refinement is warranted.

Acknowledgments

Resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute (www.vectorinstitute.ai/partners).

References

Jimmy Ba and Rich Caruana. 2014. Do Deep Nets Really Need to be Deep? In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2654–2662. Curran Associates, Inc.

Todd M. Bailey and Ulrike Hahn. 2005. Phoneme similarity and confusability. *Journal of Memory and Language*, 52(3):339–362.

Anton Batliner, Kerstin Fischer, Richard Huber, Jörg Spilker, and Elmar Nöth. 2003. How to find trouble in communication. *Speech Communication*, 40(1-2):117–143.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3(Feb):1137–1155.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2005. The AMI Meeting Corpus: A Pre-Announcement. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 28–39. Springer.

Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2015. Gated Feedback Recurrent Neural Networks. In *International Conference on Machine Learning*, pages 2067–2075.

Google Cloud. 2019. Speech-to-Text Client Libraries.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649. IEEE.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Steven L Greenspan, Raymond W Bennett, and Ann K Syrdal. 1998. An evaluation of the diagnostic rhyme test. *International Journal of Speech Technology*, 2(3):201–214.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. ISSN: 1063-6919.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kshitiz Kumar, Chaojun Liu, Yifan Gong, and Jian Wu. 2020. 1-D Row-Convolution LSTM: Fast Streaming ASR at Accuracy Parity with LC-BLSTM. *Proc. Interspeech 2020*, pages 2107–2111.

Bruce L. Lambert. 1997. Predicting look-alike and sound-alike medication errors. *American Journal of Health-System Pharmacy*, 54(10):1161–1171.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady*, volume 10, pages 707–710.

Kun Li, Xiaojun Qian, and Helen Meng. 2017. Mispronunciation Detection and Diagnosis in L2 English Speech Using Multidistribution Deep Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(1):193–207.

Ricard Marxer, Jon Barker, Martin Cooke, and Maria Luisa Garcia Lecumberri. 2016. A corpus of noise-induced word misperceptions for English. *The Journal of the Acoustical Society of America*, 140(5):EL458–EL463.

Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2011. Extensions of recurrent neural network language model. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5528–5531. IEEE.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

George A. Miller. 1954. An Analysis of the Confusion among English Consonants Heard in the Presence of Random Noise. *Journal of The Acoustical Society of America*, 26.

George A. Miller and Patricia E. Nicely. 1955. An analysis of perceptual confusions among some English consonants. *The Journal of the Acoustical Society of America*, 27(2):338–352.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Icml*.

John B. Orange, Rosemary B. Lubinski, and D. Jeffery Higginbotham. 1996. Conversational Repair by Individuals with Dementia of the Alzheimer's Type. *Journal of Speech, Language, and Hearing Research*, 39(4):881–895.

Manuel Sam Ribeiro. 2018. Parallel Audiobook Corpus. University of Edinburgh School of Informatics.

J. Dan Rothwell. 2010. *In the Company of Others: An Introduction to Communication*. New York: Oxford University Press.

Michael Sabourin and Marc Fabiani. 2000. Predicting auditory confusions using a weighted Levinstein distance. US Patent 6,073,099.

Harvey Sacks, Emanuel Schegloff, and Gail Jefferson. 1974. A Simple Systematic for the Organisation of Turn-Taking in Conversation. *Language*, 50:696–735.

Frank Seide, Gang Li, and Dong Yu. 2011. Conversational Speech Transcription Using Context-Dependent Deep Neural Networks. In *Twelfth Annual Conference of the International Speech Communication Association*.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. LSTM Neural Networks for Language Modeling. In *Thirteenth Annual Conference of the International Speech Communication Association*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Veronika Vincze, György Szarvas, Richárd Farkas, György Móra, and János Csirik. 2008. The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. *BMC Bioinformatics*, 9(11):S9.

William D Voiers, Alan D Sharpley, and Carl J Hehmsoth. 1975. *Research on Diagnostic Evaluation of Speech Intelligibility*. Research Report AFCRL-72-0694, Air Force Cambridge Research Laboratories, Bedford, Massachusetts.

Robert L. Weide. 1998. The CMU pronouncing dictionary. *The Speech Group*.

Felix Weninger, Hakan Erdogan, Shinji Watanabe, Emmanuel Vincent, Jonathan Le Roux, John R Hershey, and Björn Schuller. 2015. Speech Enhancement with LSTM Recurrent Neural Networks and its Application to Noise-Robust ASR. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 91–99. Springer.

Wayne A. Wickelgren. 1965. Acoustic similarity and intrusion errors in short-term memory. *Journal of Experimental Psychology*, 70(1):102.

Albert Zeyer, Parnia Bahar, Kazuki Irie, Ralf Schlüter, and Hermann Ney. 2019. A Comparison of Transformer and LSTM Encoder Decoder Models for ASR. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 8–15. IEEE.

Andrej Zgank and Zdravko Kacic. 2012. Predicting the Acoustic Confusability between Words for a Speech Recognition System using Levenshtein Distance. *Elektronika ir Elektrotechnika*, 18(8):81–84.

Harry Zhang. 2004. The optimality of naive Bayes. *American Association for Artificial Intelligence*, 1(2):3.

Shiyu Zhou, Linhao Dong, Shuang Xu, and Bo Xu. 2018. Syllable-Based Sequence-to-Sequence Speech Recognition with the Transformer in Mandarin Chinese. *arXiv preprint arXiv:1804.10752*.