Fluent: An AI Augmented Writing Tool for People who Stutter

Bhavya Ghai
Stony Brook University
bghai@cs.stonybrook.edu

Klaus Mueller
Stony Brook University
mueller@cs.stonybrook.edu

ABSTRACT

Stuttering is a speech disorder which impacts the personal and professional lives of millions of people worldwide. To save themselves from stigma and discrimination, people who stutter (PWS) may adopt different strategies to conceal their stuttering. One of the common strategies is word substitution where an individual avoids saying a word they might stutter on and use an alternative instead. This process itself can cause stress and add more burden. In this work, we present Fluent, an AI augmented writing tool which assists PWS in writing scripts which they can speak more fluently. Fluent embodies a novel active learning based method of identifying words an individual might struggle pronouncing. Such words are highlighted in the interface. On hovering over any such word, Fluent presents a set of alternative words which have similar meaning but are easier to speak. The user is free to accept or ignore these suggestions. Based on such user interaction (feedback), Fluent continuously evolves its classifier to better suit the personalized needs of each user. We evaluated our tool by measuring its ability to identify difficult words for 10 simulated users. We found that our tool can identify difficult words with a mean accuracy of over 80% in under 20 interactions and it keeps improving with more feedback. Our tool can be beneficial for certain important life situations like giving a talk, presentation, etc. The source code for this tool has been made publicly accessible at github.com/bhavyaghai/Fluent.

CCS CONCEPTS

• **Social and professional topics** → **People with disabilities**; • **Human-centered computing** → **Interactive systems and tools**; • **Computing methodologies** → **Active learning settings**.

KEYWORDS

Accessible computing, Stuttering, Stammering, Active learning

1 INTRODUCTION

Stuttering or Stammering is a speech disorder which is generally characterized by disfluencies like part-word repetitions eg. "I w-w-w-want a drink", prolonging a sound eg. "Ssssssssam is nice",

blocks eg. "I want a (pause) cookie", etc.[1]. Possible causes of Stuttering include Family history, differences in how the brain works during speech, etc. According to the American Speech-Language-Hearing Association (ASHA), more than 70 million people worldwide stutter[1]. It is found in all parts of the world and impacts people across culture, race, sex, age, ethnicity, etc. [4, 23, 70].

Stammering can have a profound negative impact on the personal and professional life of people who stutter (PWS). On the personal front, it can cause anxiety, frustration, embarrassment and public stigma[45]. The general public has a less positive view of such people[28] and they might face discrimination and social devaluation[6]. On the professional front, such individuals might be perceived as less capable than their peers who do not stutter. For example, physicians who stutter are perceived to be more afraid, tense, nervous and to be less mature, intelligent and competent than their peers who do not stutter[53]. Similarly, lawyers are perceived as less educated, competent and intelligent than their peers who do not stutter [54]. Also, 85% of employers think that stuttering decreases employability; and only 9% of employers think that people who stutter should be hired in a situation when two applicants are equally qualified [35]. Moreover, such individuals can also develop negative attitudes towards themselves [7].

Speech therapy can be an effective tool for handling or living with stuttering. Multiple studies have corroborated the overall positive impact of stuttering treatment on an individual[25, 68]. Having said that, there is no known cure for stuttering[32]. Moreover, access to speech therapy can be limited[63] and it might be associated with blaming and shaming[16]. As a result, many adults who stutter will continue to do so for their entire lives.

Adults who stutter are likely to have been living with their stuttering since childhood. As a person gets older, they become more conscious of their situation and learn to anticipate stuttering moments better (also known as *anticipation effect*)[21]. When an individual anticipates stuttering, they use different strategies to hide their condition which are known as *Avoidance behaviours*. This may include using fillers (e.g. 'um', 'like'), changing the feared word with its synonym (substitution), talk around the feared word (circumlocution), etc. Research has shown that avoidance behaviours are common among PWS[42] and all people who stutter use such strategies to some degree[56]. It should be noted that avoidance behaviours such as word substitution doesn't improve the underlying condition and can even have some negative effects[5, 8]. However, such tactics can help conceal stuttering with little or no observable disfluency. This might save a person from embarrassment, stress, etc. in professional settings like giving a talk, presentation, or in personal settings like dating, etc. [12].

A recent study based on a large diverse set of PWS found that the goal of the majority of the participants (69.5%) is to 'not stutter' (hide stuttering) vs 'stuttering openly' while speaking[59]. One of the factors which may impact the likelihood of stuttering is
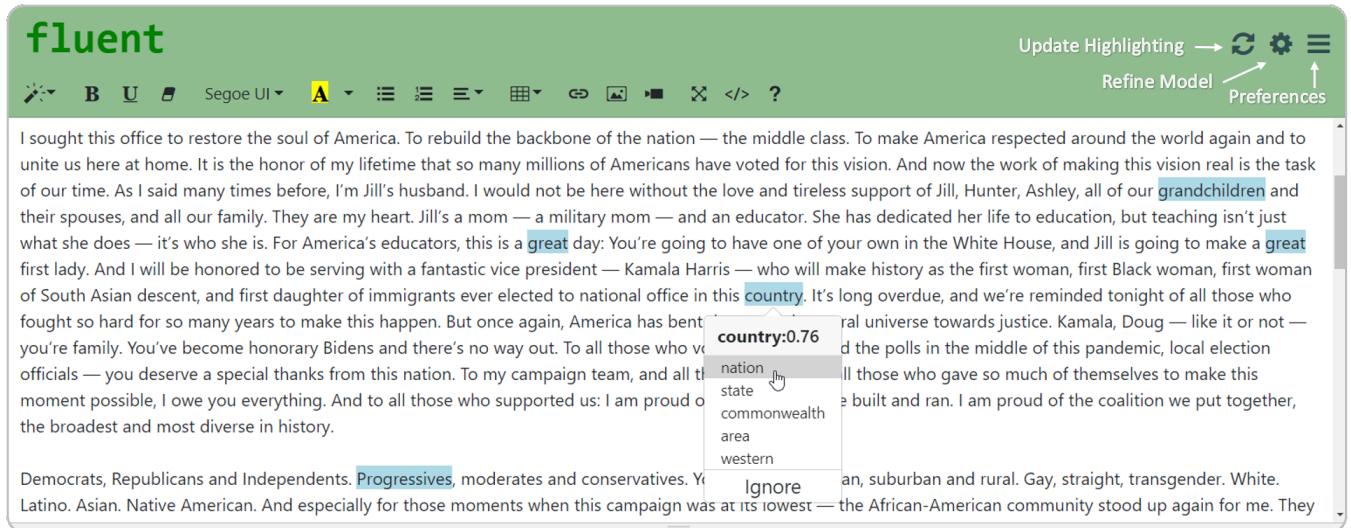
Figure 1: Visual Interface of Fluent. Words highlighted in blue are the ones which the user might find difficult to pronounce. Hovering over such words presents a set of alternatives (including Ignore option) which have similar meaning but might be easier to pronounce. In the above picture, the user hovers over the word 'country' and the tool presents a set of alternatives namely, nation, state, commonwealth, area, etc. Buttons on the top right corner allows the user to provide explicit feedback (Refine Model) and provide a set of words which they find easy/difficult to pronounce (Preferences).

phonological patterns[26]. In other words, PWS are likely to stutter on some words more than others. To prevent oneself from stuttering, one has to identify which words they might struggle with and then think of a way to manage it. This process itself can take time, effort and can cause additional stress [44]. In this work, we leverage recent advancements in AI such as phonetic embeddings to reduce this burden and help PWS hide their stuttering.

We present *Fluent*, a novel machine in the loop writing tool for assisting PWS with writing scripts, which they can speak more fluently (minimize the number of stuttering events). Given a piece of text, our tool helps identify words that a person might struggle pronouncing (trigger words). Such words are highlighted in the user interface in real time. For each of such words, our tool provides a set of alternatives which have similar meaning but might be easier to pronounce. The user can simply hover over any of the highlighted words to choose from the set of alternatives (see Figure 1). Since each user can have different requirements, our tool evolves itself via user feedback to provide better personalized support over time. Fluent can be used for writing speeches, scripts, dialogues which might be used by anchors, politicians, actors, etc. This might not only save time but also enhance their confidence and ultimately impact their performance. The primary contributions of this work are as follows:-

- We devise a novel method to identify words an individual might struggle pronouncing.
- We design and implement a new writing tool 'Fluent' which embodies our approach to identify trigger words and suggests suitable alternatives.
- We evaluate our system by measuring its ability to identify trigger words for 10 simulated users.

2 BACKGROUND AND RELATED WORK

2.1 AI for Stuttering

Existing literature at the intersection of AI and stuttering focuses on building machine learning systems to identify and classify different types of disfluencies like Blocks, Prolongations[17], Sound Repetitions [47], Interjections, etc. in speech utterances[9]. Such systems are typically trained on speech samples which are annotated for different kinds of disfluencies [2, 17, 18, 36, 57]. Other approaches have leveraged data based on facial muscle movements[13], breathing patterns[64], brain activity[30], etc. The goal of such systems is to assist speech language pathologists (SLPs) during the stuttering assessment phase. During the assessment phase, the SLP counts the number and type of stuttering events in speech. This process can be tedious and subjective [37]. Automated ML systems can perform this task and help save time while providing objective results.

Overall, this promising space seems under-explored as a lot of the work in this area pivots around a single problem i.e., stuttering detection. In this work, we have taken a different route of developing assistive writing technology for PWS by leveraging AI based methods. Unlike previous work, our system relies on self reported textual data to learn more granular phonetic patterns that an individual might struggle pronouncing. Moreover, our system utilizes user feedback to provide more personalized feedback.

2.2 AI based Writing Tools

The landscape of AI based writing tools typically comprises of Natural Language Processing (NLP) based software systems aimed at improving productivity[10], creativity[20], inclusion[58], etc. *Grammarly*[33] is a popular AI writing assistant which provides real time suggestions for fixing grammatical errors, improved word
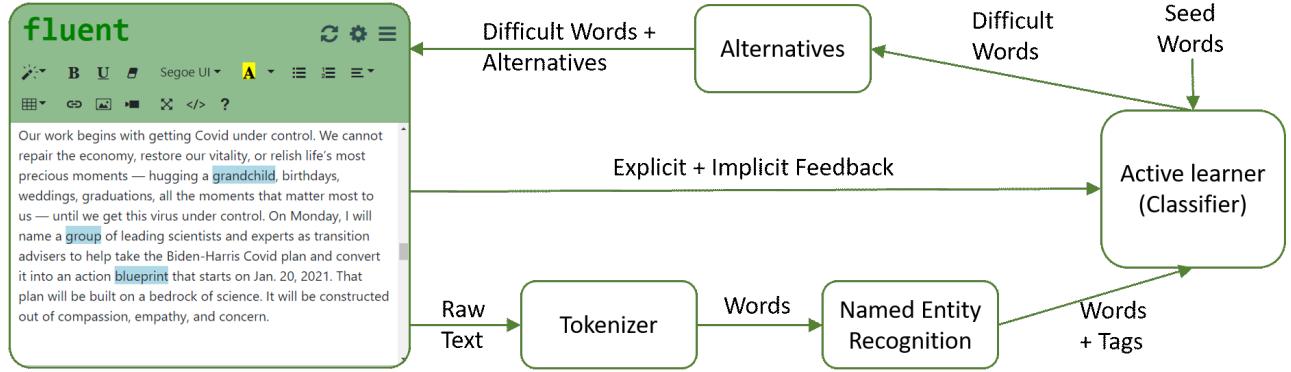
Figure 2: Workflow of Fluent

choice, refining tone, etc. *Textio*[58] is another AI powered writing tool which helps with hiring content. It suggests word level changes for writing inclusive and effective job descriptions. Similarly, there are other tools meant for a specialized task or audience. For example, *FLOW*[10] is an interactive writing assistant for people who learn english as a foreign language, *Creative Help*[48] and *LISA*[49] help with story writing, *SWAN*[34] helps with scientific writing, etc.

There has been some research on developing smart authoring tools for people with special needs. For example, Wu et al.[67] developed a writing tool powered by Neural Machine Translation model to empower people with Dyslexia. There are also writing tools meant for people with visual impairments[24, 65], Sign Language Users[29], etc. However, little to no attention has been paid towards people with speech disorders. In this paper, we have taken a small step to fill this gap by developing the first smart writing tool for people who stutter (PWS).

2.3   Active Learning
Supervised machine learning (ML) models require large amounts of labeled training data to provide good results. In many ML settings, unlabeled data points are abundant but labeling them can be time consuming and expensive[69]. For such cases, Active learning (AL) serves as a viable learning paradigm as it focuses on training effective ML models using a minimum number of labeled training instances[50]. AL achieves this by intelligently selecting data samples from a pool of unlabeled data which are then labeled by the oracle (e.g., a human annotator)[38]. The ML model is retrained iteratively for every new set of labeled data points as they come in. This process continues for a predefined number of iterations or until the annotation budget lasts[3]. Here, the data points can be selected using different sampling strategies like Uncertainty sampling[38], Query by committee sampling[19, 52], Hierarchical sampling[14], QUIRE[27], etc. Active learning has been found to work well for different applications[22, 46] including Text classification[61], Named entity recognition[11], etc. However, it comes with its own set of practical challenges[31, 51]. Research has shown that its benefits might not generalize reliably across models and tasks[39]. In this work, we are trying to leverage AL in the context of stuttering. More specifically, we are investigating if AL can help learn the unique phonetic patterns that an individual might struggle pronouncing.

3   FLUENT
3.1   Design Goals
We have identified the following design goals based on the existing literature on stuttering, interactive systems[66] and the personal experiences of an author of this paper who stutters:

G1. Identify Difficult Words: Given a piece of text, our tool should identify words which a given individual might find hard to pronounce. Here, each individual might struggle with different sounds, syllables, etc. Our goal is to build a generic tool which can learn the individual requirements of each user to provide personalized support.

G2. Alternatives: Given a word which might be hard to pronounce, our tool should provide a set of alternatives which have similar meaning and can can be easily pronounced by the individual it is targeted to.

G3. Interface Design: The Interface should be designed such that it is easy to latch onto (intuitive), accessible and should shield the user from underlying technical details (minimalist). It should provide the desired functionality while ensuring minimal lags to ensure smooth user experience.

3.2   Identifying Difficult to Pronounce Words
An intuitive solution to classify words based on their pronunciation can be to use acoustic word embeddings. Such embeddings provide a vector representation for a given speech signal. However, we might get different embeddings for the same word pronounced by people from different gender, age, etc. [62]. Moreover, dealing with sound can be more computationally expensive (defeating G3). Hence, we used Phonetic embeddings [43] based on the CMU Pronouncing Dictionary which is independent of speaker bias. Phonetic embeddings map each word to its corresponding vector representation based on the constituting phonemes. Words with similar pronunciation will be closer to each other in the embedding space.

In the beginning, the user is asked to list a set of at least 5 words which they find easy and difficult to pronounce, respectively (see Fig. 3). The larger the number of words, the better. Such words are also referred as seed words (see Figure 2). Here, we are expecting the user to have a certain level of self awareness about their condition. Each word in either lists is mapped to its corresponding phonetic
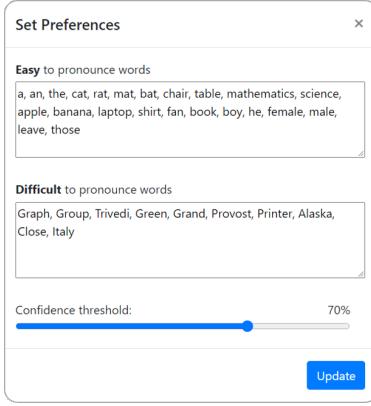
Figure 3: User Preferences. The user can provide details on which words they find easy/difficult to pronounce.

embedding[43]. Thereafter, we train an off the shelf binary SVM model (Active learner) to classify between easy and difficult words. For a given word, this trained model returns a numerical score between 0 and 1 which represents the likelihood that the word is hard to pronounce. We use this classifier over each word in the text editor and highlight words whose probability of being difficult is greater than a threshold. By default, the threshold value is 0.7. The user is free to change this threshold. A higher threshold might result in more false negatives and a lower threshold might lead to more false positives. Here, we have used a spacy[1] based tokenizer to extract words from the raw text in the interface.

3.3 Adapting to Personalized needs

The initial classification model trained over few training instances can be suboptimal. To refine the model further and better adapt it to personalized needs, we utilize explicit and implicit feedback from the user. Implicit feedback is gathered based on which option is selected from the dropdown menu when hovering over a highlighted word. If the user chooses the 'Ignore' option, we add the highlighted word to our list of *easy* words. Else we add it to the list of *hard* words and add the alternative word chosen to the *easy* list. Thereafter, we retrain the model over the updated word lists.

For explicit feedback, we ask the user to indicate if a specific word is difficult or not (see Figure 4). Here, the chosen word is selected from the pool of all unlabeled words in the phonetic embedding about which the current SVM model is most uncertain about. More specifically, we use entropy based uncertainty sampling[50] to select the next word. Based on the user provided label, the word is added to an existing list of *easy* or *hard* words. Thereafter, the model is retrained based on the updated word lists and the next word with highest predictive uncertainty is chosen. This iterative process might lead to fast convergence of the SVM model while requiring minimum number of labeled instances.

3.4 Finding Alternative Words

To find suitable alternatives for a given word, we first generate a list of words which have similar meaning. Thereafter, we discard
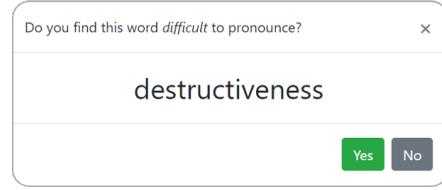
Figure 4: Explicit Feedback: Query for refining Active learning classifier

those synonyms which might be difficult to say as per the current SVM classifier. One possible option to generate synonyms is using online dictionaries like Thesaurus. Such dictionaries provide high quality synonyms but are limited in their coverage i.e., they do not provide synonyms for many words. Another possible approach is to use nearest neighbors as synonyms in word embeddings like word2vec[40]. This approach provides broad coverage but the quality of synonyms suffer. In this work, we have used the DataMuse API[2] which uses multiple online dictionaries, WordNet, Word2Vec and other databases to yield good quality synonyms for most words. Moreover, we identify words which represent names, places, dates, etc. using Named Entity Recognition from the spacy[1] package. Our tool just highlights such words and leaves it to the user to deal with such cases; it doesn't generate alternatives for such words.

3.5 Interface Design

Fluent is implemented as a web application using python based web framework *Flask*. The visual interface (see Fig.1) is built using javascript based open source library *Summernote*. This makes Fluent easily accessible using a web browser across different platforms without needing any third party software or specialized computing resources. The default visual interface is designed to look like a generic rich text editor. Additional features pertaining to stuttering popup only upon clicking buttons on the top right of the interface. This is to ensure easy flow of ideas without getting distracted by too many options. For highlighting a word, we took inspiration from popular writing tools like Textio[58] which also uses a different background color to highlight a word. To display the set of alternative words, we have used a popup mechanism populated with alternatives which appears right below the word being hovered over. Such a design choice is implemented to mimic spell checkers and other tools like Grammarly[33] which most people might have interacted with at some point. This might help users to quickly latch on to our interface without needing additional training.

4 EVALUATION

We evaluate our system in term of its ability to identify trigger words for 10 hypothetical users who interact with the system in a specific manner.

User Profiles. We intended to evaluate our system on a diverse set of realistic user profiles with varying degrees of stuttering. So, we modeled 4 different user profiles based on self reported data from stuttering communities on facebook/reddit and one from

---

[1]https://spacy.io/

[2]https://www.datamuse.com/api/
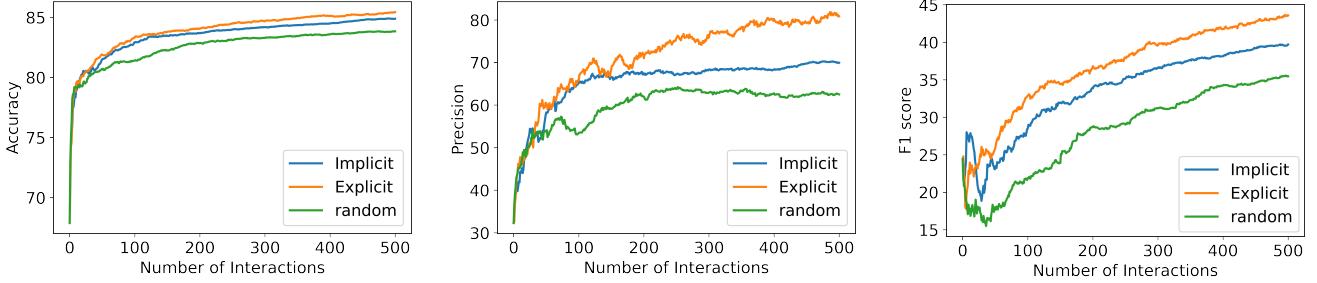
Figure 5: Plots showing the mean values for Accuracy, Precision & F1 Score for 10 simulated users across 500 interactions.

the personal experiences of an author of this paper who stutters. Here, each user profile is defined by a phonetic pattern that they find difficult to pronounce and their corresponding seed words (see Supplementary material). For example, User 1 struggles to pronounce words which start with a consonant followed by a 'r' sound. To model severe stuttering for the last 5 profiles, we used a combination of two or three phonetic patterns from the top 5 profiles. For example, User 10 struggles with 3 phonetic patterns i.e., words starting with 'st' or 'fl', words with an 'r' or 'l' letter at the second place and words with a 'sc' or 'ch' sound anywhere in the word.

Data. We used 2,467 TED talks transcripts[3] data which contains 57k unique words to evaluate our tool. We computed the ground truth (true label) for each word across all users based on their respective phonetic pattern. For each unique word, we got 10 binary labels corresponding to each user. With the ground truth available, we split the 57k words into train and test data in the ration 75:25. We evaluate our classifier on how well it can predict the true label for the test data.

Simulation. At the beginning, each user provides 5 easy and 5 difficult words based on their unique condition. We train a SVM classifier based on these words. Thereafter, we have simulated two scenarios where the user provides only explicit or implicit feedback. This is to investigate how well each feedback mechanism works individually and relative to each other. In reality, a user can choose to provide both forms of feedback. For reference, we have also added a third scenario where we label words randomly. To simulate each interaction for the implicit feedback scenario (confidence threshold = 0.1.), each user searches for the first highlighted word among the TED dataset and interacts with it. Here, the user chooses the 'Ignore' option if the highlighted word is a false positive. Otherwise, the user chooses the first alternative suggested by the system. For explicit feedback, we assume that each user will provide the correct response for all queries. After each explicit/implicit feedback (interaction), word lists are updated and the model is retrained. Thereafter, we measure metrics like precision, accuracy and f1 score corresponding to each user over the test dataset. Higher values for such metrics indicates better personalized support.

Results. Figure 5 shows the mean scores for accuracy, precision and F1 score for 10 users across 500 interactions. We can see an overall positive trend for both forms of feedback across different

[3]Dataset link: https://www.kaggle.com/rounakbanik/ted-talks?select=transcripts.csv

metrics. In under 20 interactions, the classifier reached a mean accuracy of over 80% for both forms of feedback (random classifier will yield 50%). This shows that both forms of feedback are effective in enhancing the performance of our classifier. It is interesting to observe that explicit feedback is more effective than implicit feedback and random labeling across different metrics. It should be noted that an implicit feedback can potentially add two data points (if the highlighted was actually a trigger word) compared to a single data point for explicit feedback per interaction. This suggests that active learning can significantly accelerate the learning process in the context of stuttering. So, it is advisable that the user provides explicit feedback whenever possible to accelerate the learning process.

Implicit feedback might be a bit slower but it may be a more natural and non-intrusive way to provide feedback. To help users choose a threshold wisely (see Figure 3), we explored the relation between confidence threshold and F1 score for the implicit feedback scenario (see Figure 6). Our experiment suggests that F1 score can vary significantly for different threshold values and a lower threshold might generally yield higher F1 score.

Overall, our experiments demonstrate that our approach of using phonetic embeddings combined with active learning provides promising results. Our tool can effectively learn the personalized needs of different users within a short span of time.

5 DISCUSSION, LIMITATIONS AND FUTURE WORK

Intended Use. Fluent is designed to assist PWS get through certain important life situations like giving a talk. In the process, it might make the person more self-aware about their condition by helping them discover trigger words which they were previously unaware of. It might also enrich their vocabulary by suggesting alternatives. However, it should be noted that word substitution is a just a coping mechanism and it doesn't improve the underlying condition. Our tool just helps at concealing the underlying condition better. Avoiding behaviors can be empowering[12] but they can also have some negative effects. For example, research has shown that people who try to conceal stuttering report lower levels of self-esteem and quality of life[5, 8]. Users are advised to visit a speech language pathologist for proper personalized treatment.

Design. This work ventures into a previously unexplored territory of building a smart text editor for PWS. So, there are no existing design guidelines to build such a system. In this work, we have emulated the general design principles of smart text editors
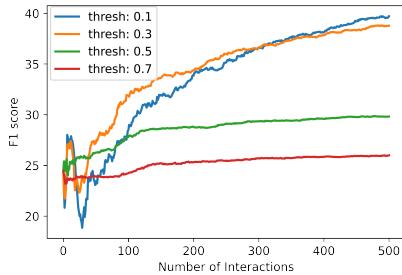
Figure 6: Plot showing how F1 score varies for different thresholds. Larger the threshold, lower the mean F1 score.

like Grammarly and relied upon the personal experiences of an author of this paper who stutters to make certain design choices. Such design decisions may/mayn't generalize to the PWS at large. Future work might involve PWS to evaluate the design decisions made in this work and devise a set of comprehensive design guidelines to inform the development of such tools in the future.

Efficacy. We have focused on evaluating the learning capability of our system from a machine learning perspective. Our promising results are based on 10 simulated users who interacted with the system in a specific predefined fashion. In the real world, people might struggle on varied sets of words and and might interact with the system in different ways. The next step will be to perform a comprehensive evaluation from a human centered perspective. It will be interesting to conduct an empirical study with PWS and investigate the usability, utility and effectiveness of Fluent relative to a plain text editor. Future work might explore the utility of this tool for speech therapy and also measure the impact of this tool on an individual's confidence and stress levels while speaking.

Substitution. Each word in the English language has its own subtle meaning. So, it might not be possible to substitute a word completely. However, we have tried to suggest similar meaning words by leveraging knowledge from multiple dictionaries, WordNet[41], word2vec embedding[40], etc. Future work might incorporate contextualized word embeddings like BERT[15], etc. to further refine the set of alternatives for a given word. It should be noted that words representing immutable categories like names, places, dates, etc. can't be substituted. Our tool just highlights such cases and leaves it to the user to deal with them.

Target Users. Fluent requires the user to provide seed words in the beginning and assumes that the subsequent user feedback is mostly accurate. This requires the user to have a certain level of self awareness about one's condition. Moreover, if one suffers from acute stammering, then one might need to substitute multiple words in a sentence which might break the semantic structure of the sentence. So, our tool might not work well for such people. Future work might explore the utility of our tool for non-native speakers and other speech disorders like lisp. For example, people who have a lisp might find certain sounds like "s" hard to pronounce. So, they might struggle on words like misses, session, aesthetics, etc. Similarly, native french speakers find words like thorough, through, clothes, etc. difficult because 'th' sound doesn't exist in french. It

will be interesting to see how well our tool can identify words with such underlying phonetic pattern.

Vocabulary. Fluent can analyze and possibly highlight only those words which are present in the phonetic embedding. In our case, the phonetic embedding has more than 116k unique words which is substantially larger than an average person's active vocabulary size of ~20k words. So, our tool should be able to deal with most words used by an average person. However, it might not contain domain specific jargon. Moreover, our tool can't deal with numbers say 64, abbreviations say NY and symbols say $ unless they are written in worded form like Sixty four, New York, etc. Future work might might try to deal with such cases and extend support for other languages (not just english).

Granularity. Fluent processes running text by breaking it down into words and then analyzes each one of them individually. It is possible that an individual might pronounce a word (say 'juror') well on its own but might struggle when the same word occurs in a specific context/sentence (say 'rural juror'). Our current system doesn't capture such cases as it analyzes each word on its own without considering its context. Future work might account for such cases by analyzing word n-grams (contiguous sequence of words) apart from just individual words.

User Feedback. Fluent elicits binary user feedback i.e., whether a word is difficult to pronounce or not. Such a feedback is less cognitively taxing on the user and can also be gathered in a non-intrusive fashion (Implicit feedback). However, the real situation might be more subtle. For e.g., a person might struggle on a word sometimes but not always. To cater to such situations, our tool can be easily modified to elicit to a more nuanced (continuous) feedback instead of a discrete binary label. For eg., we can ask the user to rate their confidence in the binary label or to rate the difficulty of a word on a n-point likert scale. Such nuanced feedback can be incorporated by existing active learning frameworks for more effective learning[55]. Here, we should also study and consider the kinds of feedback PWS naturally want to provide[22]. Apart from severity, stuttering behavior can also vary with time and situation[60]. Future work should also try to capture such variability.

6 CONCLUSION

We have presented *Fluent*, a smart text editor meant for people who stutter. Fluent embodies a novel adaptive, interactive and iterative technique to identify words which an individual might struggle pronouncing and proposes suitable alternatives. Our experiments show promising results and corroborate this problem as a possible research direction. We hope this work will encourage other researchers to work on this important and under-explored area. In the long run, we hope our work might inspire popular tools like MS PowerPoint, MS Word, etc. to add accessibility features for PWS.

ACKNOWLEDGMENTS

REFERENCES

[1] Accessed August 2020. Stuttering. (Accessed August 2020). https://www.asha.org/stuttering/ ASHA: American Speech-Language-Hearing Association.

[2] Sadeen Alharbi, Madina Hasan, Anthony JH Simons, Shelagh Brumfitt, and Phil Green. 2020. Sequence labeling to detect stuttering events in read speech. *Computer Speech & Language* 62 (2020), 101052.

[3] Michael Altschuler and Michael Bloodgood. 2019. Stopping active learning based on predicted change of f measure for text classification. In *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*. IEEE, 47–54.

[4] Gavin Andrews, Susan Hoddinott, Ashley Craig, Pauline Howie, Anne-Marie Feyer, and Megan Neilson. 1983. Stuttering: A review of research findings and theories circa 1982. *Journal of speech and hearing disorders* 48, 3 (1983), 226–246.

[5] MP Boyle. 2016. Relations between stuttering disclosure and self-empowerment in adults who stutter. In *Poster presented at the American Speech-Language-Hearing Association, Annual Convention, Philadelphia, PA*.

[6] Michael P Boyle. 2018. Enacted stigma and felt stigma experienced by adults who stutter. *Journal of Communication Disorders* 73 (2018), 50–61.

[7] Michael P Boyle and Alison N Fearon. 2018. Self-stigma and its associations with stress, physical health, and health care satisfaction in adults who stutter. *Journal of Fluency Disorders* 56 (2018), 112–121.

[8] Michael P Boyle, Kathryn M Milewski, and Carolina Beita-Ell. 2018. Disclosure of stuttering and quality of life in people who stutter. *Journal of fluency disorders* 58 (2018), 1–10.

[9] Lim Sin Chee, Ooi Chia Ai, and Sazali Yaacob. 2009. Overview of automatic stuttering recognition system. In *Proc. International Conference on Man-Machine Systems, no. October, Batu Ferringhi, Penang Malaysia*. 1–6.

[10] Mei-Hua Chen, Shih-Ting Huang, Hung-Ting Hsieh, Ting-Hui Kao, and Jason S Chang. 2012. FLOW: a first-language-oriented writing assistant system. In *Proceedings of the ACL 2012 System Demonstrations*. 157–162.

[11] Yukun Chen, Thomas A Lasko, Qiaozhu Mei, Joshua C Denny, and Hua Xu. 2015. A study of active learning methods for named entity recognition in clinical text. *Journal of biomedical informatics* 58 (2015), 11–18.

[12] Christopher Dominick Constantino, Walter H Manning, and Susan Naomi Nordstrom. 2017. Rethinking covert stuttering. *Journal of Fluency Disorders* 53 (2017), 26–40.

[13] Arun Das, Jeffrey Mock, Henry Chacon, Farzan Irani, Edward Golob, and Peyman Najafirad. 2020. Stuttering Speech Disfluency Prediction using Explainable Attribution Vectors of Facial Muscle Movements. *arXiv preprint arXiv:2010.01231* (2020).

[14] Sanjoy Dasgupta and Daniel Hsu. 2008. Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*. ACM, 208–215.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[16] Jill E Douglass, Catherine Kennedy, and Kaitlyn Smith. 2020. Speech-Language Therapy Experiences Across the Life Span of an Individual Who Covertly Stutters: A Thematic Analysis. *Perspectives of the ASHA Special Interest Groups* 5, 6 (2020), 1441–1453.

[17] Iman Esmaili, Nader Jafarnia Dabanloo, and Mansour Vali. 2017. An automatic prolongation detection approach in continuous speech with robustness against speaking rate variations. *Journal of medical signals and sensors* 7, 1 (2017), 1.

[18] Fabio Fassetti, Ilaria Fassetti, and Simona Nisticò. 2019. Learning and detecting stuttering disorders. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*. Springer, 319–330.

[19] Yoav Freund, H Sebastian Seung, Eli Shamir, and Naftali Tishby. 1997. Selective sampling using the query by committee algorithm. *Machine learning* 28, 2-3 (1997), 133–168.

[20] Jonas Frich, Lindsay MacDonald Vermeulen, Christian Remy, Michael Mose Biskjaer, and Peter Dalsgaard. 2019. Mapping the landscape of creativity support tools in HCI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–18.

[21] Mauricio A Garcia-Barrera and Jason H Davidow. 2015. Anticipation in stuttering: A theoretical model of the nature of stutter prediction. *Journal of fluency disorders* 44 (2015), 1–15.

[22] Bhavya Ghai, Q. Vera Liao, Yunfeng Zhang, Rachel Bellamy, and Klaus Mueller. 2021. Explainable Active Learning (XAL): Toward AI Explanations as Interfaces for Machine Teachers. *Proc. ACM Hum.-Comput. Interact.* 4, CSCW3, Article 235 (Jan. 2021), 28 pages. https://doi.org/10.1145/3432934

[23] Barry Guitar. 2013. *Stuttering: An integrated approach to its nature and treatment*. Lippincott Williams & Wilkins.

[24] Tomáš Hanakovič and Marek Nagy. 2006. Speech recognition helps visually impaired people writing mathematical formulas. In *International Conference on Computers for Handicapped Persons*. Springer, 1231–1234.

[25] Carl Herder, Courtney Howard, Chad Nye, and Martine Vanryckeghem. 2006. Effectiveness of behavioral stuttering treatment: a systemic review and meta-analysis. *Contemporary Issues in Communication Science and Disorders* 33, Spring (2006), 61–73.

[26] Peter Howell, James Au-Yeung, Scott J Yaruss, and Kevin Eldridge. 2006. Phonetic difficulty and stuttering in English. *Clinical linguistics & phonetics* 20, 9 (2006), 703–716.

[27] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. 2010. Active learning by querying informative and representative examples. In *Advances in neural information processing systems*. 892–900.

[28] Lloyd M Hulit and Lauralee Wirtz. 1994. The association of attitudes toward stuttering with selected variables. *Journal of fluency disorders* 19, 4 (1994), 247–267.

[29] Nedelina Ivanova and Olle Eriksen. 2012. A Bilingual Bimodal Reading And Writing Tool For Sign Language Users. *Universal Learning Design, Linz 2012* (2012), 79.

[30] Jing Jiang, Chunming Lu, Danling Peng, Chaozhe Zhu, and Peter Howell. 2012. Classification of types of stuttering symptoms based on brain activity. *PloS one* 7, 6 (2012), e39747.

[31] Jean-François Kagy, Tolga Kayadelen, Ji Ma, Afshin Rostamizadeh, and Jana Strnadova. 2019. The Practical Challenges of Active Learning: Lessons Learned from Live Experimentation. *arXiv preprint arXiv:1907.00038* (2019).

[32] Joseph S Kalinowski and Tim Saltuklaroglu. 2005. *Stuttering*. Plural Publishing.

[33] Laksnoria Karyuatry. 2018. Grammarly as a Tool to Improve Students' Writing Quality: Free Online-Proofreader across the Boundaries. *JSSH (Jurnal Sains Sosial dan Humaniora)* 2, 1 (2018), 83–89.

[34] Tomi Kinnunen, Henri Leisma, Monika Machunik, Tuomo Kakkonen, and Jean-Luc Lebrun. 2012. SWAN - Scientific Writing AssistaNt: A Tool for Helping Scholars to Write Reader-Friendly Manuscripts. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics* (Avignon, France) *(EACL '12)*. Association for Computational Linguistics, USA, 20–24.

[35] Joseph F Klein and Stephen B Hood. 2004. The impact of stuttering on employment opportunities and job performance. *Journal of fluency disorders* 29, 4 (2004), 255–273.

[36] Tedd Kourkounakis, Amirhossein Hajavi, and Ali Etemad. 2020. FluentNet: End-to-End Detection of Speech Disfluency with Deep Learning. *arXiv preprint arXiv:2009.11394* (2020).

[37] Deborah Kully and Einer Boberg. 1988. An investigation of interclinic agreement in the identification of fluent and stuttered syllables. *Journal of fluency disorders* 13, 5 (1988), 309–318.

[38] David D Lewis and William A Gale. 1994. A sequential algorithm for training text classifiers. In *SIGIR'94*. Springer, 3–12.

[39] David Lowell, Zachary C Lipton, and Byron C Wallace. 2018. Practical obstacles to deploying active learning. *arXiv preprint arXiv:1807.04801* (2018).

[40] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[41] George A Miller. 1995. WordNet: a lexical database for English. *Commun. ACM* 38, 11 (1995), 39–41.

[42] Bill Murphy, Robert W Quesal, and Hope Gulker. 2007. Covert stuttering. *Perspectives on Fluency and Fluency Disorders* 17, 2 (2007), 4–9.

[43] Allison Parrish. 2017. Poetic sound similarity vectors using phonetic features. In *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference*.

[44] Laura W Plexico, Walter H Manning, and Heidi Levitt. 2009. Coping responses by adults who stutter: Part I. Protecting the self and others. *Journal of fluency disorders* 34, 2 (2009), 87–107.

[45] Jane E Prasse and George E Kikano. 2008. Stuttering: an overview. *American family physician* 77, 9 (2008), 1271–1276.

[46] Maria E Ramirez-Loaiza, Manali Sharma, Geet Kumar, and Mustafa Bilgic. 2017. Active learning: an empirical study of common baselines. *Data mining and knowledge discovery* 31, 2 (2017), 287–313.

[47] KM Ravikumar, Balakrishna Reddy, R Rajagopal, and H Nagaraj. 2008. Automatic detection of syllable repetition in read speech for objective assessment of stuttered disfluencies. *Proceedings of world academy science, engineering and technology* 36 (2008), 270–273.

[48] Melissa Roemmele and Andrew S Gordon. 2015. Creative help: A story writing assistant. In *International Conference on Interactive Digital Storytelling*. Springer, 81–92.

[49] Rushit Sanghrajka, Daniel Hidalgo, Patrick Chen, and Mubbasir Kapadia. 2017. Lisa: Lexically intelligent story assistant. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 13.

[50] Burr Settles. 2009. *Active learning literature survey*. Technical Report. University of Wisconsin-Madison Department of Computer Sciences.

[51] Burr Settles. 2011. From theories to queries: Active learning in practice. In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*. JMLR Workshop and Conference Proceedings, 1–18.

[52] H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. 1992. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 287–294.

[53] Franklin H Silverman and Tamara A Bongey. 1997. Nurses' attitudes toward physicians who stutter. *Journal of Fluency Disorders* 1, 22 (1997), 61–62.

[54] Franklin H Silverman and Kathryn K Paynter. 1990. Impact of stuttering on perception of occupational competence. *Journal of Fluency Disorders* 15, 2 (1990), 87–91.

[55] Jinhua Song, Hao Wang, Yang Gao, and Bo An. 2018. Active learning with confidence-based answers for crowdsourcing labeling tasks. *Knowledge-Based Systems* 159 (2018), 244–258.

[56] C Woodruff Starkweather. 1987. *Fluency and stuttering.* Prentice-Hall, Inc.

[57] Izabela Świetlicka, Wiesława Kuniszyk-Jóźkowiak, and Elżbieta Smołka. 2013. Hierarchical ANN system for stuttering identification. *Computer Speech & Language* 27, 1 (2013), 228–242.

[58] Textio. [Online]. The augmented writing platform. https://textio.com/ (accessed 4 April 2021).

[59] Seth E Tichenor and J Scott Yaruss. 2019. Group experiences and individual differences in stuttering. *Journal of Speech, Language, and Hearing Research* 62, 12 (2019), 4335–4350.

[60] Seth E Tichenor and J Scott Yaruss. 2021. Variability of stuttering: Behavior and impact. *American Journal of Speech-Language Pathology* 30, 1 (2021), 75–88.

[61] Simon Tong and Daphne Koller. 2001. Support vector machine active learning with applications to text classification. *Journal of machine learning research* 2, Nov (2001), 45–66.

[62] Lisa van Staden and Herman Kamper. 2020. Improving unsupervised acoustic word embeddings using speaker and gender information. In *2020 International SAUPEC/RobMech/PRASA Conference.* IEEE, 1–6.

[63] Sarah Verdon, Linda Wilson, Michelle Smith-Tamaray, and Lindy McAllister. 2011. An investigation of equity of rural speech-language pathology services for children: A geographic perspective. *International Journal of Speech-Language Pathology* 13, 3 (2011), 239–250.

[64] Bruno Villegas, Kevin M Flores, Kevin José Acuña, Kevin Pacheco-Barrios, and Dante Elias. 2019. A novel stuttering disfluency classification system based on respiratory biosignals. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).* IEEE, 4660–4663.

[65] Mirza Muhammad Waqar, Muhammad Aslam, and Muhammad Farhan. 2019. An intelligent and interactive interface to support symmetrical collaborative educational writing among visually impaired and sighted users. *Symmetry* 11, 2 (2019), 238.

[66] Austin P Wright, Omar Shaikh, Haekyu Park, Will Epperson, Muhammed Ahmed, Stephane Pinel, Duen Horng Chau, and Diyi Yang. 2021. RECAST: Enabling User Recourse and Interpretability of Toxicity Detection Models with Interactive Visualization. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (2021), 1–26.

[67] Shaomei Wu, Lindsay Reynolds, Xian Li, and Francisco Guzmán. 2019. Design and evaluation of a social media writing support tool for people with dyslexia. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.* 1–14.

[68] J Scott Yaruss. 2010. Assessing quality of life in stuttering treatment outcomes research. *Journal of fluency disorders* 35, 3 (2010), 190–202.

[69] Xiaojin Jerry Zhu. 2005. Semi-supervised learning literature survey. (2005).

[70] Gerald Zimmermann, Sven Liljeblad, Arthur Frank, and Charlotte Cleeland. 1983. The Indians have many terms for it: Stuttering among the Bannock-Shoshoni. *Journal of Speech, Language, and Hearing Research* 26, 2 (1983), 315–318.