Wait-Learning: Leveraging Wait Time for Second Language Education

Wait-Learning: Leveraging Wait Time for Second Language Education

Carrie J. Cai[1] **Philip J. Guo**[1,2] **James R. Glass**[1] **Robert C. Miller**[1]

[1]MIT CSAIL
Cambridge, MA USA
{cjcai, glass, rcm}@mit.edu

[2]University of Rochester
Rochester, NY USA
pg@cs.rochester.edu

ABSTRACT

Competing priorities in daily life make it difficult for those with a casual interest in learning to set aside time for regular practice. In this paper, we explore *wait-learning*: leveraging brief moments of waiting during a person's existing conversations for second language vocabulary practice, even if the conversation happens in the native language. We present an augmented version of instant messaging, WaitChatter, that supports the notion of wait-learning by displaying contextually relevant foreign language vocabulary and micro-quizzes just-in-time while the user awaits a response from her conversant. Through a two week field study of WaitChatter with 20 people, we found that users were able to learn 57 new words on average during casual instant messaging. Furthermore, we found that users were most receptive to learning opportunities immediately after sending a chat message, and that this timing may be critical given user tendency to multi-task during waiting periods.

Author Keywords

Wait-learning; Micro-learning; Second language learning.

ACM Classification Keywords

H.5.2. User interfaces: User-centered design; K.3.1. Computer Uses in Education: Computer-assisted instruction

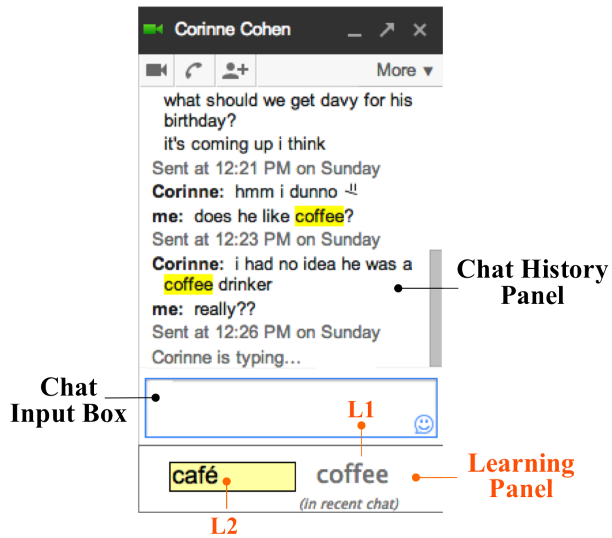


Figure 1: WaitChatter presents vocabulary exercises in the *learning panel* while the user awaits an IM response. Here, the user is being quizzed on a word and must enter the second language (L2) translation given the native language (L1) prompt. The word is highlighted in the chat history because it appeared in the context of the conversation.

INTRODUCTION

Learning a second language requires significant time and effort on a recurring basis. Living in a country where the language is spoken is often necessary to reach fluency, yet many language learners do not have the time to dedicate years of their lives to immersive instruction. Even for those who attempt to learn informally, the busyness of daily life makes it difficult to schedule regular time for practice. Learners must maintain the *executive motivation* [12] necessary to study the second language on a repeated basis.

Despite the struggle to find time for learning, there are countless moments in a day that go wasted, due to suboptimal scheduling or necessary waiting [19]. Recent work on *micro-learning* has explored ways to distribute traditional language study into many micro moments throughout a person's daily life [16]. In this work, we extend micro-learning and introduce the notion of *wait-learning*. The novelty of this approach lies in the targeted use of time that users would ordinarily spend waiting. The proposed benefit is that learning during wait time will minimize the likelihood that learning is perceived by learners as intrusive or time-consuming.

Due to inherent difficulties in predicting whether someone is waiting based purely on tracking their activities [19], we turn to an activity that necessarily involves waiting *within* the activity itself. Instant messaging (IM) offers a powerful opportunity for wait-learning due to its semi-asynchronous nature. Because messages being typed are unseen by the conversant and can be revised before being sent, a user must often wait for a brief moment in anticipation of a response, with no guarantee that the other party will in fact reply [28].

CHI 2015, April 18–23, 2015, Seoul, Republic of Korea.

ACM 978-1-4503-3145-6/15/04 ...$15.00.
http://dx.doi.org/10.1145/2702123.2702267

In this paper, we explore the notion of wait-learning with WaitChatter (Figure 1), an extension of instant messaging that presents vocabulary exercises while the user awaits an IM response. The interaction is asymmetric, meaning that the user can engage in learning without any knowledge of this activity by the other party. Based on findings from a wizard-of-oz exploratory study [7], we designed and implemented the WaitChatter system as an extension of Google Chat in the Chrome web browser. Through a two-week field study in which 20 participants used WaitChatter on their personal computers, we find that users are most receptive to learning in the moments after sending a chat message. We also show that wait-learning can facilitate learning. On average, participants learned 57 new vocabulary words over the course of two weeks, while exchanging on average 170 chats per day.

The main contributions of this paper are:

- the central idea of *wait-learning*, an extension of micro-learning that leverages wait time for education
- a system that implements wait-learning by presenting learning exercises at automatically-detected moments of waiting during instant messaging
- a naturalistic evaluation that suggests users are most receptive to wait-learning in the moments after sending a chat message

BACKGROUND AND RELATED WORK

Our research on wait-learning extends existing work on micro-learning to use wait time for education. It also builds upon prior research on instant messaging and interruptions.

Micro-Learning and Contextual Micro-Learning

Based on growing evidence that spaced exposure [11] and continued exposure [30] to second language vocabulary result in greater learning gains, recent work on *micro-learning* [16] has explored methods to distribute learning into small units throughout a person's day-to-day life. Micro-learning has largely been implemented in the form of *contextual micro-learning*, through mobile applications that provide opportunities to learn contextually relevant vocabulary based on surrounding objects [5], location [10, 14] and text [29].

Spaced exposure has been posited to benefit learning, even in the absence of context [11]. Due to low interactivity, certain instructional elements that can be reasonably learned in isolation may be more appropriate for micro-learning. For example, the study of vocabulary, computer terminology, and chemical symbols are posited to impose a lower cognitive load in comparison to learning algebra, which has more interactive parts [26].

Prior work on contextual micro-learning has tended to focus more on *what* the user is learning rather than *when* to present these learning opportunities. However, several researchers have aimed to make use of the transition time between tasks to present learning opportunities. Lernschoner [16] activates a learning program whenever the learner's computer screen becomes idle. MicroMandarin [14] and Vocabulary Wallpaper [10] leveraged moments while users are visiting certain locations to present location-sensitive vocabulary, though even these studies tended to focus more on the content delivered rather than the timing of those deliveries.

Micro-Waiting during Instant Messaging

Prior research suggests that even micro-diversions during frequent activities could be perceived as disruptive if the user feels that regular tasks are being delayed [29]. Beyond micro-learning, our work on wait-learning specifically targets moments when users would ordinarily spend waiting.

Research on *mobile micro-waiting* suggests it is difficult to use activity information alone to predict idleness, possibly because users may be more receptive to content delivered during fleeting intervals of idle time *within* an ongoing activity [19]. In one study on in-home interruptability, participants rated watching television as both a high and low engagement activity depending on their goals [27].

During instant messaging, waiting often occurs within the activity itself. Unlike email and face-to-face communication, instant messaging conversations range from rapid exchanges to long periods of time passing between messages [23]. Due to its semi-asynchronous nature, users frequently use waiting breaks between conversation turns to attend to other informal activities [18]. Prior work has explored ways to reduce the attention cost of multi-tasking, such as by predicting interruptability [2, 15] and examining how IM notifications affect other tasks [9]. While these examples aim to decrease time wasted, our work instead makes use of this time by encouraging users to engage in learning while waiting for a response.

Attention and Interruption

Although wait-learning aims to leverage idle time, previous work on interruptability suggests that the detailed manner and timing of delivery can still cause significant differences in task performance and levels of frustration [1]. Because our minds dynamically allocate and release resources throughout task execution [20], the timing of information delivery relative to a user's ongoing task may affect interruption cost [4].

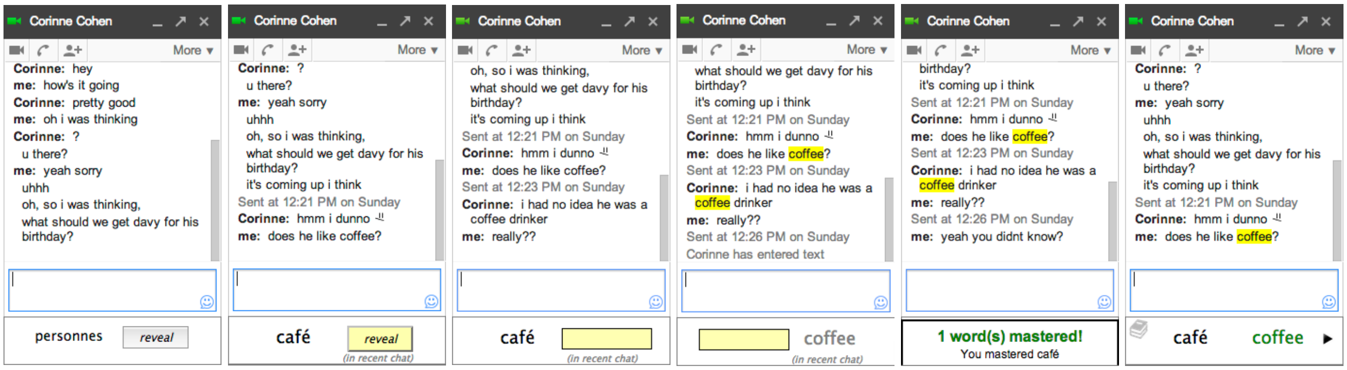
Because decreases in workload tend to be larger at boundaries corresponding to larger chunks of a task [4, 22], a system that presents information during IM should differentiate between the many boundaries that might exist, favoring those that represent more salient breaks in workflow [4]. In one study on face-to-face communication, users processed additional information best when presented with small batches of information, and when the user was not speaking [24].

WAITCHATTER DESIGN AND IMPLEMENTATION

In order to explore and empirically validate wait-learning as it applies to instant messaging, we developed WaitChatter (Figure 1) as an extension of Google Chat that runs in the Web browser when a Gmail page is in view. The client is packaged as a Chrome extension with a Python-based server backend.

User Interface Design

A central design challenge is the small amount of space available for designing interactions. Because instant messaging is intended to be the main task, with learning as a secondary



(a) Study Mode for a non-contextual word. (b) Study Mode for a contextual word. (c) Quiz Mode at easy level (translate to L1). (d) Quiz Mode at difficult level (translate to L2). (e) Mastery panel slides up when word is learned. (f) Clicking the arrow fetches the next exercise.

Figure 2: WaitChatter teaches words during automatically detected wait-moments. Here are components of the user interface.

benefit, user interface elements should not demand a substantial context switch away from the IM conversation.

Based on findings from an exploratory study [7], the learning panel is situated directly below the chat input box to minimize the visual and motor effort of switching between chatting and learning activities. To avoid user concern over whether chat conversants can view WaitChatter content, we keep the exercises within the learning panel, but highlight a keyword inside the chat history if it is selected and presented for learning.

Vocabulary exercises appear and remain on the learning panel for 10 seconds, during which the user can either do the exercise or ignore it by doing nothing. If the user has not interacted with the exercise for 10 seconds, it fades away. To minimize disruption, the learning panel remains present at all times regardless of whether an exercise is present. After the user completes an *initial* exercise, he can fetch a *follow-up* one by clicking the right arrow (Figure 2f) or hitting the right arrow key on the keyboard. This functionality allows the user to continuously learn more words during longer wait times.

A vocabulary word is displayed in *study mode* (Figure 2a, 2b) the first time it is presented for learning, and in *quiz mode* (Figure 2c, 2d) during subsequent exposures. In study mode, the L2 (second language) vocabulary word is shown alongside a button which the user can click to view the L1 (native language) translation. Once a learner reveals the new word, the user is asked to indicate whether he or she already knew that word. If not, the word is added to the learner's *vocabulary list*, a growing list of words that the user did not already know but has been exposed to via WaitChatter. In quiz mode, the user is shown the L2 word and provides the L1 translation if the exercise is at the *easy* level (Figure 2c), or the reverse if the exercise is at the *difficult* level (Figure 2d). Users can submit a blank response if they don't remember the word.

Vocabulary

WaitChatter aims to support the learning of second language vocabulary. We limit our implementation to nouns because, in addition to being closely tied to the meaning of a conversation, nouns are also typically acquired before verbs and are easier to translate automatically because they are less context-dependent. A related system [29] also focused on nouns for similar reasons.

Findings from a prior study indicate that contextual and non-contextual vocabulary serve complementary roles in learning, balancing needs across language levels, and that users may benefit from a combination of the two [14]. Although that study examined context with respect to location relevance, in WaitChatter the IM conversation itself provides a ripe opportunity for in-context learning. The nouns in WaitChatter are either drawn from a built-in word list (*non-contextual*, Figure 2a), or selected on-the-fly from words used by either conversant in the IM conversation (*contextual*, (Figure 2b). *Non-contextual* words could in theory be drawn from a number of sources, such as a word bank seeded by the learner or teacher.

For each chat message exchanged during an IM conversation, WaitChatter determines whether an adequate *contextual* word exists. First, it identifies nouns in the chat message using the Senna part-of-speech tagger [8]. Then, each noun is translated on-the-fly using Google Translate. To maximize the chances that the word is translated correctly in context, WaitChatter sends both the word and the entire chat message to Google Translate, and considers an L1/L2 pair *accepted* only if the L2 word appears in both translation results. To mitigate against inaccurate results, a dictionary icon (Figure 2f) is displayed which the user can click to see the word's dictionary definition. For privacy reasons, WaitChatter logs only the length of a chat message and the L1/L2 pair displayed in an exercise, but not the content of the chat message itself.

Because exercises can only be displayed during appropriate wait-learning moments (described below), WaitChatter keeps a running set of *accepted* contextual L1/L2 pairs, for which the L1 word is still within the visible part of the chat history (*viewport*) but not yet displayed for learning. Once the word scrolls off the viewport, WaitChatter discards the L1/L2 pair from being considered for learning.

Vocabulary Scheduling Algorithm

WaitChatter uses the Leitner schedule [17] for scheduling the order of learning exercises. The Leitner schedule is based on the principle of *spaced repetition*: given that humans exhibit a negatively exponential forgetting curve [13], repetitions should occur at increasingly spaced intervals so that they are encountered just as they are about to be forgotten.

We define each flashcard in the system as an L1/L2 pair. WaitChatter maintains a set of five unlearned flashcards and a *correct count* for each flashcard, which represents the number of correct responses to that flashcard. This count is incremented when the learner answers the flashcard correctly and decremented if not. Flashcards with a correct count of *n* are displayed every *n*th Leitner session, so that better known cards are reviewed less frequently. In WaitChatter, flashcards are displayed at the easy level when the correct count is below three, and at the difficult level otherwise. A flashcard is "learned" and retired when its correct count reaches four (Figure 2e), opening up a slot for a new card to be added.

In WaitChatter, the Leitner algorithm was modified to enable a combination of contextual and non-contextual words. First, if an opportunity arises for a new card to be added, WaitChatter picks a contextual word if one is available. If no contextual word can be found, WaitChatter shows the next unused non-contextual word at random from the same Leitner session. To capitalize on contextual opportunities, any word already on the vocabulary list that re-appears in the context of the conversation is prioritized to be displayed at the next wait opportunity, regardless of the Leitner state.

During pilot studies, some users indicated a preference for seeing new words over repeatedly seeing old ones. To maximize engagement, we rotate words within the same Leitner session whenever a word is ignored so that learners are unlikely to ignore the same stale word consecutively. Furthermore, if a new slot becomes available during a *follow-up* exercise, WaitChatter will display the newly added flashcard at the next *initial* exercise than showing it immediately, and instead display the next flaschard in the Leitner schedule. Since follow-up exercises are requested by the user whereas initial exercises are not, we display new flashcards during initial exercises with the intent of maximizing engagement.

Detecting Waiting Opportunities

Results from an exploratory study [7] identified two situations in which a user may be waiting during an instant messaging conversation: 1) waiting for the conversant to start responding, and 2) waiting for the conversant to finish responding. Figure 3 shows these two types of waiting opportunities in the flow of a typical conversation.

The first case, which we name *i_sent*, occurs after a user has sent a chat message and is waiting to see whether his conversant will respond. Because a common IM behavior is to type a sequence of short chat messages as part of one conversational turn [18, 21], an exercise that is naively delivered immediately after a chat is sent may interrupt a follow-up message that the user is in the midst of composing. For this reason, WaitChatter waits for an additional amount of *hesitation time* after a message is sent, and subsequently triggers a learning exercise only if the user has not typed more. We used 1.5 seconds of hesitation time to balance against the user tendency to

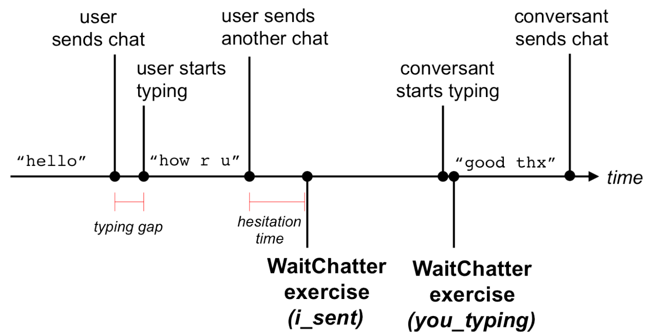


Figure 3: Detection of waiting opportunities in WaitChatter.

leave the chat window after sending a message. According to a prior study [2], the probability that the message window is still in focus is approximately 60% after 1 second, and drops to approximately 50% within 5 seconds.

The second case, which we name *you_typing*, occurs when the conversant has started typing a response but has not yet sent the message. In Google Chat and other similar instant messaging applications, users see an indicator (i.e. "Corinne is typing...") which signals that the conversant has started typing. WaitChatter triggers an exercise when the indicator appears in the user's chat window and the user is not typing. In both *i_sent* and *you_typing*, the exercise is only triggered if the cursor focus is still inside the chatbox.

The creation of an exercise requires server-side interaction, a round-trip process that results in a small but unpredictable delay. WaitChatter will cancel an exercise before it is displayed if it detects that the user has typed between the time that the browser requested an exercise and received a server response.

EVALUATION

To evaluate WaitChatter and the extent to which wait-learning can help second language acquisition, we ran a two-week field study in which participants used WaitChatter in Google Chat on their personal computers during their normal instant messaging activities.

The questions our study sought to answer are:

- Learning: To what extent can users learn vocabulary using WaitChatter?
- Timing: What are the best times to present learning exercises for the purpose of wait-learning?

Vocabulary

For ease of user recruitment, our implementation of WaitChatter teaches Spanish and French, but could be extended to other languages. Non-contextual vocabulary were drawn from high frequency English nouns as measured in the British National Corpus.[1] The words were translated to Spanish and French using Google Translate, after which two native speakers manually reviewed the word list for inaccurate translations and highly ambiguous words. The final non-contextual word lists consisted of 446 words in each language.

[1] http://www.natcorp.ox.ac.uk/

Procedure

Participants first met with a researcher to have WaitChatter installed on their personal computers. They were then given a walkthrough of how WaitChatter could be used. Participants used WaitChatter as they pleased for the next two weeks. During the study, WaitChatter prompted participants to indicate whether or not they already knew a word the first time it appeared. Known words were not added to the user's vocabulary list. At the end of two weeks, participants returned to complete a post-study questionnaire, semi-structured interview, and vocabulary quiz. The quiz tested all vocabulary the user had been exposed to (but didn't already know) while using WaitChatter, and was divided into two parts: participants translated from L1 to L2 in the *recall* quiz, and from L2 to L1 in the *recognition* quiz. Within each part, the order of questions was randomized. Users were asked not to guess.

Timing Variation

To better understand how the timing of exercises may affect the learner's capacity to engage in learning, we exposed each participant to two versions of our application: the *detected_wait* version uses the *i_sent* and *you_typing* waiting opportunities as described above. The *random* version displays prompts at random whenever WaitChatter determines that a user has been actively instant messaging. We define a user's instant messaging activity as *active* if the Gmail page is in focus, has had keyboard or mouse activity within the last 30 seconds, and contains at least one open chat window which had keyboard activity within the last 5 minutes. Because a user may be waiting for an instant messaging response while engaged in nearby tasks on the same page, we consider moments when the Gmail page is in focus even though the chatbox is not as viable candidates for wait-learning, so long as the user is active as defined above. We also require at least one chatbox to be open because an open chatbox indicates the likely presence of an ongoing conversation [2].

Each participant used the *detected_wait* and *random* versions on alternating days. To ensure that users are exposed to WaitChatter prompts at approximately equal frequencies on the *detected_wait* and *random* versions, the desired frequency on a *random* condition day was determined by calculating the cumulative frequency, which is the number of exercises shown per time active, from the user's previous *detected_wait* condition days. On *random* days, this desired frequency was used every second the user was active to probabilistically determine whether an exercise should be shown.

We examined three metrics: whether the learner responded to the exercise, the time taken to respond, and users' subjective impression of how well the exercises integrated into the flow of their activities. We define response time as the time between a prompt being displayed and the user's cursor focusing into the answer box. To capture subjective impressions, users were asked to complete a daily survey with two 7-point Likert scale questions: 1) "In the past day, [WaitChatter] exercises appeared at good moments within the flow of my daily activities", and 2) "I enjoyed using [WaitChatter] today." The survey was sent via email every evening, instructing users to complete it once they finish chatting at the end of the day.

Participants

21 participants were recruited by emails sent through university department, dorm, and foreign language course email lists. We selected only those who were regular users of Google Chat in the web browser, desired to learn or were currently learning Spanish or French, and were not traveling for more than 2 days of the two-week study period. One participant was dropped midway through the study because they stopped instant messaging and completing the daily surveys after the sixth day. Participants were given a $30 gift card for their time, and also entered into a raffle for one $100 gift card.

The 20 participants who completed the study included 12 males and 8 females, ages 19 to 35 (mean=25.5). They consisted of mostly undergraduate and graduate students (17 out of 20), as well as two alumni working in industry and one research scientist. Eleven chose to learn French and nine learned Spanish. Ten users had prior formal education in the language ranging from elementary school (2) and middle or high school (6) to university-level classes (2). Eight of the participants had studied the language informally through using language learning software, traveling in a foreign country, or talking to friends. Six participants had never studied the language before, either formally or informally. The participants typically use Google Chat on their computers "Several times an hour" (9) or "Several times a day" (11), mostly for social reasons or to chat with labmates about research.

RESULTS AND DISCUSSION

Overall, we observed 47393 instant messages exchanged by the 20 participants, who communicated with a total of 249 friends. Users exchanged an average of 170 chats per day.

Learning

In just two weeks of casual usage, participants were on average able to recall 57 new words, equivalent to approximately four words per day. Participants were on average exposed to 106.7 words, 87.7 (sd=64.8) of which they didn't already know. Among those, approximately 40-60% of words were cognates. In post-study quizzes, users translated 57.1 words (66%) correctly to L2 and 80.2 words (92%) correctly to L1. In quiz translations to L2, 15% of wrong answers appeared to be spelling errors or near-misses. The user who was exposed to the most new words (256) translated 161 correctly to L2 and 232 correctly to L1. The most infrequent chatter (55 chats per day) learned 17 new words. These results suggest that down time during instant messaging can serve as a viable channel for learning, at least for bite-sized information.

Some participants wished old words they had already mastered could be revived after a few days so that they wouldn't be as easily forgotten. Because there was no limit on the number of times a learner could continuously fetch more exercises, it was possible for a user to "learn" a flashcard in a single sitting. A spacing algorithm that incorporates temporal effects, such as that described in [25], could be used in the future to improve long term retention.

Usage Patterns

During post-study interviews, users reported behavior that resembled episodes in which they were learning while waiting.

For example, participants said they tended to complete exercises *"while waiting for people to respond," "while the other person is thinking,"* or *"when it was unpredictable how long my friend would be gone for."*

Users indicated that they were most likely to interact with WaitChatter during a *continuous but casual* IM conversation. They were least likely to engage with exercises if they were having a particularly time-sensitive conversation or if the nature of the conversation was serious or work-related. As one user put it, *"The best times are when I'm talking continuously with one person, but we're not having a very heated conversation. Just like hi, how are you, and when the material is more light."* Thus, WaitChatter usage seemed to occur during periods of "outeraction" [23], when people communicate for the purpose of maintaining social connection and awareness, rather than for specific information exchange.

High-usage participants indicated that they frequently used the "fetch more" feature to complete a long sequence of exercises if the conversation was particularly sporadic: *"At some points I needed to wait for the other person to respond. The longer they take, the more words I would go through."* Others were more likely to use WaitChatter to complete only one or two exercises in the transition time *between* chatting and a primary task. Because instant messaging is itself a common break-time activity, IM functioned as a bootstrap for learning when the main task did not require high mental effort: *"If I'm doing something else at the same time like packing and stuff, I might hear the ping, read the reply, do a word or two, get up and go back to what I was doing."*

Overall, we found that people tend to fill their wait time by doing casual tasks, and that wait-learning in many ways served as a timely replacement for those alternative downtime activities: *"There were definitely times when I would keep clicking the arrow because whatever's on TV was really boring."* Another user remarked, *"Maybe I'm just chatting and looking at Facebook. Instead I would use WaitChatter because it's more productive."*

Responsiveness to Timing Conditions

We first measure the amount of time between messages during IM conversations to understand the extent to which learning exercises can be feasibly completed during this time. We found that the time between the user sending a message and receiving a message from the conversant (Figure 4) exhibits a distribution with a peak and a long tail, similar to responsiveness distributions reported in a prior IM study [2]. This intermessage distribution includes instances in which the conversant has started composing a message before the user has sent his message. The time taken to complete an exercise was short (median=1.83 seconds), well within the median intermessage time (11 seconds). However, because intermessage time is most commonly short (mode=4 seconds), particularly during conversations with frequent exchanges, it is reasonable that learning exercises be lightweight.

To understand how the user's ability to respond to exercises could be affected by different timing conditions, we looked at the following measures: whether the user responded to the exercise, and the time taken to respond, as defined in the Timing Variation section above. For each user, data on the first *detected_wait* condition day and the first *random* condition day were excluded from analysis to avoid novelty effects. Furthermore, because follow-up exercises are requested by the user whereas initial exercises are not, we focus our analysis only on initial quiz exercises.

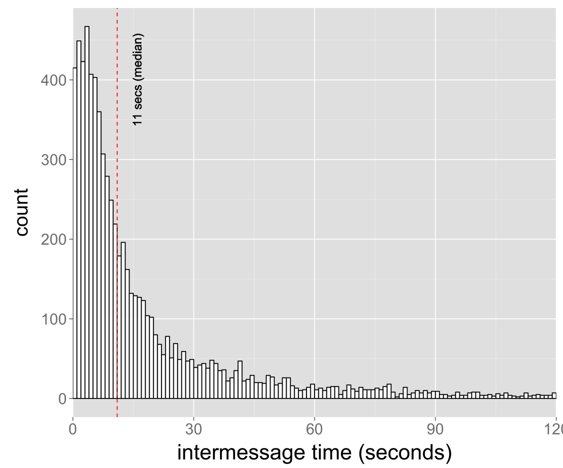


Figure 4: Histogram of intermessage time: the time between the user sending a message and receiving a message from the conversant. Bin size is 1 second.

Because *i_sent* and *you_typing* exercises occurred only while the user had cursor focus inside the chatbox, we subdivided exercises on the *random* condition days into *random_inside*, when the chatbox had focus, and *random_outside*, when the chatbox did not have focus. We compared *i_sent* and *you_typing* only to *random_inside* trials. In all cases, the user had been actively instant messaging as defined in the Timing Variation section above. Table 1 displays a summary of conditions.

Whether the learner responded

We found that 43.5% of the exercises received a response in the *random_inside* condition, whereas only 31.2% received a response in the *random_outside* condition. A generalized linear mixed effects analysis with the Timing condition (*random_inside* and *random_outside*) as the fixed effect and Participant as a random effect found that *random_outside* deliveries were significantly less likely to receive a response ($p<0.0001$). This analysis excludes one participant who did not receive any *random_outside* exercises. Consistent with a prior study [2] which found that chat window focus may be a strong indicator for chat responsiveness, our results imply that this applies to learning interactions as well.

Comparing *i_sent*, *you_typing*, and *random_inside*, we found that response rate was highest for *i_sent* (49.1%), followed by *random_inside* (44.5%) and *you_typing* (41.2%). In a generalized linear mixed effects analysis, p-values were 0.0085 (*i_sent*>*you_typing*), 0.0498 (*i_sent*>*random_inside*, and 0.397 (*you_typing*=*random_inside*), of which only the

Condition	Day Shown	Condition Requirement
i_sent	detected_wait (odd days)	1.5 seconds after user sends a chat, provided he has not started typing again
you_typing	detected_wait (odd days)	“[conversant name] is typing...” indicator appears and user is not typing
random_inside	random (even days)	Chat window is in focus
random_outside	random (even days)	Chat window is not in focus

Table 1: Summary of conditions.

first passed the Bonferroni-corrected threshold of 0.0167 (*i_sent*>*you_typing*).

The relatively low response rate in the *you_typing* condition could be due to a number of factors. *you_typing* moments may occur when users are visually focused on another screen, even if the cursor is focused inside the chatbox. Conversely, *random_inside* prompts that happen to be delivered mid-typing are likely to be noticed, and could conceivably be attended to before the exercise disappears. Moreover, as chat messages tend to be short [21], users may already be receiving a response by the time they are able to attend to the exercise.

Time to respond

As shown in Figure 5, response time for the *random_inside* condition (mean=3973 ms, sd=792) was faster than that of the *random_outside* condition (mean=4888, sd=1282). A paired t-test found this difference to be significant ($F(1,16)=13.24$, $p<0.005$). This analysis excludes three participants who did not respond to any *random_outside* exercises. The longer time to respond in the *random_outside* condition could be due to the mental cost of switching from another activity, or the extra distance traveled to reach the answer box.

Comparing *i_sent*, *you_typing*, and *random_inside* (Figure 6), we found that *i_sent* had the fastest response time (mean=3628, sd=637), followed by *random_inside* (mean=4009, sd=792), and *you_typing* (mean=4209, sd=969). A repeated measures ANOVA showed a significant effect of condition ($F(2,38) = 4.00$, $p<0.05$). Post-hoc Bonferroni tests revealed that learners were significantly quicker to engage with prompts delivered during the *i_sent* condition than those presented during *random_inside* ($p<0.05$). Users were also quicker to engage with *i_sent* than *you_typing*, a difference that was marginally significant ($p=0.05$).

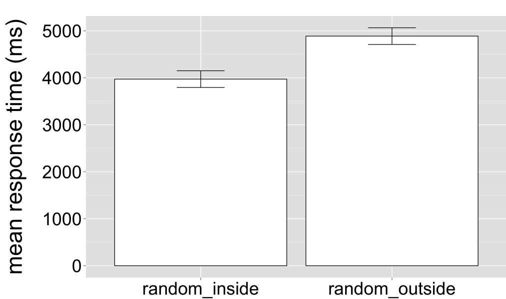


Figure 5: Mean response times for *random_inside* and *random_outside* exercises. Error bars show SE of the mean.

Results suggest that exercises appearing during *you_typing* were not easier to process while conversing, despite occurring at a detected wait moment. It is possible that, during *you_typing*, users are already in the midst of planning their next message, or concentrating on what their friend will say in their response. The mental resources released may be small due to a large carryover of information being actively maintained in short-term memory, which is characteristic of low-level task boundaries or non-boundaries [4]. Conversely, *i_sent* occurs almost immediately after the user has finished typing, and may thus be more consistent with moments of lower mental workload because it occurs between the completion of one subtask and the beginning of the next [22].

During the study, it appeared that in the *random_inside* condition, many users responded quickly to an exercise even if it appeared while they were typing. We thus examined response times depending on the number of keystrokes remaining in the message being composed (Figure 7). We split these instances into two equally sized groups, and found that the response time of prompts arriving just before the user sent a chat (within the last 8 keystrokes) is lower (median=2216ms, sd=2248) than those arriving earlier (median=3446ms, sd=2175). This result suggests that exercises arriving immediately prior to message completion may receive fast engagement, possibly because the user is almost done typing and has mentally queued up the exercise.

Lastly, we evaluate the 1.5 second hesitation time that WaitChatter used to ensure the user was not typing a followup message before it delivered a learning exercise in the *i_sent* condition. Due to missing log data for some users, we report preliminarily on the seven users for whom we had a full set of millisecond-accurate data, including all times they pressed the enter key. We limit our analysis to events where the user

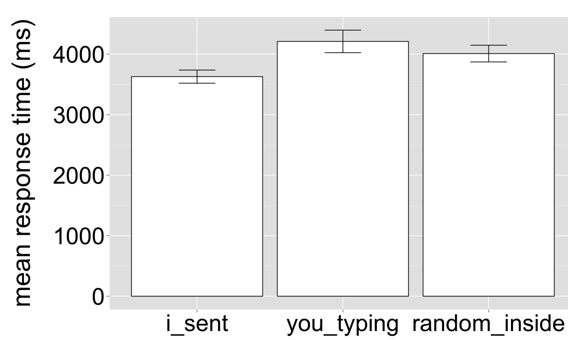


Figure 6: Mean response times for *i_sent*, *you_typing*, and *random_inside*. Error bars show SE of the mean.

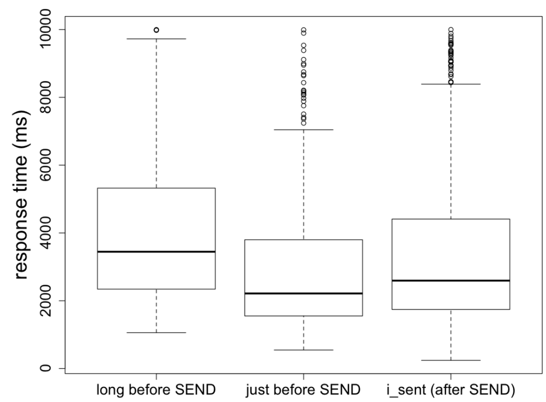


Figure 7: Box plots of response times to exercises arriving just before a chat is sent ($\leq$ 8 keystrokes left), compared to longer before a chat is sent ($>$ 8 keystrokes left).

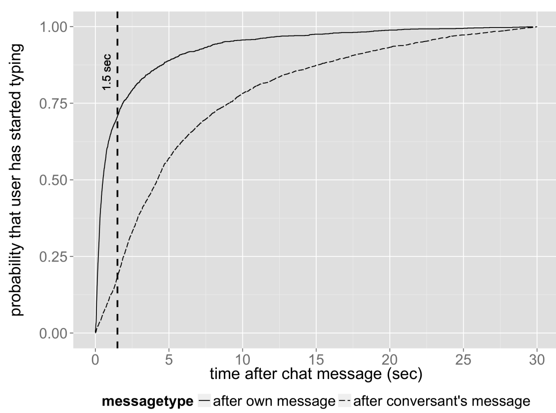


Figure 8: If the user has started typing within 30 seconds of sending a chat, there is a 70% chance that this re-typing occurs within 1.5 seconds. In contrast, if the user starts typing within 30 seconds after *receiving* a chat, there is only a 18% that it occurs within 1.5 seconds.

re-started typing within 30 seconds, based on prior research findings that 28-30 seconds is a typical amount of time between conversational turns [3]. Among these instances, 70% occurred within WaitChatter's hesitation threshold of 1.5 seconds, making it a reasonable estimate (Figure 8). Nevertheless, the hesitation threshold ought to be balanced against the user tendency to leave the chat window after sending a message. A more lenient implementation of WaitChatter might set the hesitation threshold to be even lower than 1.5 seconds.

Perception of Time and Disruption

Because WaitChatter specifically makes use of wait time, we were interested in how users perceived the time and effort they spent on learning, as compared to alternative learning methods. Here, we report on results from the post-study questionnaires and interviews related to this question.

Perception of Time Spent

As shown in Figure 9, responses to 7-point Likert scale questions (1=strongly disagree, 7=strongly agree) indicated that users found WaitChatter very enjoyable (mean=6.15). They also felt that they would continue using WaitChatter if they could (mean=6.15), and would engage in vocabulary practice more frequently than they would otherwise (mean=6.6). On the last question, 15 out of 20 participants submitted a rating of 7.

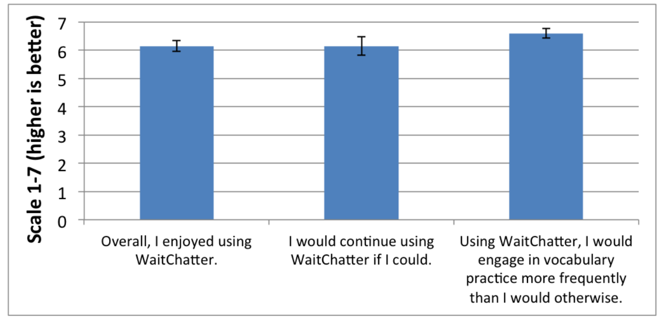


Figure 9: Post-study questionnaire results, with standard error bars.

During interviews, nearly every participant expressed that WaitChatter felt less time-consuming compared to other channels of learning because they did not need to set aside time for learning. As one user stated, "*The key thing is that I didn't feel like I was taking extra time out of my day to dedicate to learning vocabulary words. It was just sort of time that would be wasted otherwise.*" Another compared it to typical break-time activities: *"Some people play Angry Birds, but for me, I would play with [WaitChatter]. At least I'm learning some French words."*

Many contrasted WaitChatter to language courses and software, which they felt required a conscious effort to schedule time for learning. One person commented, "*With Duolingo you have to think 'I have to go do this now', whereas with [WaitChatter] it's already done for you, spoonfed to you.*" Another said, "*With this I never had to make time or put away things to the future. Whereas learning from Rosetta Stone, you have to schedule time.*" Most who had used vocabulary-learning mobile applications indicated that they eventually gave up, citing time as a major factor.

Several users noted that the little time required to complete a WaitChatter exercise ironically encouraged them to interact more with it overall. One person described the low time commitment as follows: *"It's just like, here, literally just take 2 seconds!"* Another appreciated the regularity of exposure: *"You're just constantly getting new words. It might be a slower rate overall [compared to classes], but it's neat in the aspect that you're always chugging along, learning new vocabulary."* These comments suggest that user engagement could hinge more on the perceived than the actual time spent.

Perception of Disruption

Overall, participants felt that WaitChatter integrated well into their daily lives because they already instant message regularly: "*I already have gmail open and I'm always chatting with people anyway.*" Furthermore, the ease of access was key to their frequent usage: "*It being so close to the chatbox* *made me do it more. If it was a separate thing it wouldn't be as easy – an extra click away is a large amount of effort.*"

In the 7-point Likert scale questions sent daily (1=strongly disagree, 7=strongly agree), users on average felt that the exercises "appeared at good moments within the flow of my daily activities" (mean=5.45, sd=1.05) and that they "enjoyed using WaitChatter today" (mean=5.61, sd=1.02). A Wilcoxon Signed-rank test found no significant difference between user ratings on *detected_wait* versus *random* condition days, for either question (p=0.71 and p=0.57).

During interviews, users indicated that they did not notice systematic differences in the timing of exercises, and that the appearance of exercises almost never felt disruptive because they always had the option to ignore them: *"It was just a matter of choice. Subconsciously I would see that a word has appeared. Sometimes it would pique my interest and I would look at it, but if not I just wouldn't look at it so it wasn't really disrupting anything."* These findings are consistent with the notion that interruptions which do not occlude the primary task are perceived to be less mentally demanding and less annoying [6]. Unlike pop-up notifications, the learning panel was in a persistent self-allocated space, making it potentially less distracting even in cases when timing was sub-optimal.

Some users reported feeling frustrated when they could not attend to the exercise in time because they were still typing a long message. Hence, while users did not perceive the timing of appearances to be intrusive, they may be less tolerant of the premature disappearance of an exercise.

LIMITATIONS AND FUTURE WORK

One limitation of this study design was that the frequency of surveys may have been too coarse-grained to capture more transient impressions. We sent the survey only once a day to prevent the frequency of the survey itself from negatively impacting user experience. Future studies may consider sampling user impressions closer in time to the user's actual encounter with learning exercises. In addition, we alternated timing conditions on a daily basis to balance the frequency of exercises. While it is possible that more frequent switching could better mask the conditions, users expressed during the interviews that they did not detect a noticeable pattern in the timing of exercises.

Furthermore, our implementation was limited to vocabulary learning. Some users wished the system could additionally teach sentence structures and phrases. It remains to be seen whether more complex structures can be taught in a way that still preserves the bite-sized nature of wait-learning. For example, a more sophisticated system could present more complex exercises when users actively fetch more exercises, indicating that they are open to learning more, or when the waiting period is particularly long.

DESIGN IMPLICATIONS

The timing and manner in which learning components appear during waiting opportunities can have important implications for actual end-user use. In the following sections, we consider four implications related to the use of wait time for education.

Engaging Prior to Context-Switching

Learning exercises should be delivered immediately at the start of or prior to waiting, rather than times that are well into the waiting period. Because users are likely to attend to other tasks while waiting, they may be less likely to engage in learning once they have already shifted attention to other default waiting activities. Results from our user study indicate that educational prompts delivered right after the user has sent a message tend to receive the fastest engagement, and that this may be true even if exercises are presented within the last few keystrokes while the user is still typing.

Multiplexing the Learning Component

To ensure a non-intrusive user experience, learning components should appear in a way that allows the user to be sufficiently informed of the arrival of a learning opportunity without losing interactivity with their main task. Participants in our study found WaitChatter to be non-intrusive even when exercises appeared amidst typing, citing their degree of control and ability to ignore prompts as primary reasons.

Extending Engagement with Optional Continuations

Because the system could be imperfect in predicting when a user is interruptable, educational prompts should serve as an entryway to further learning. If a prompt is well-timed and the user chooses to interact, this opportunity should be capitalized to encourage continuation of learning. In WaitChatter, the right-arrow enabled users to fetch more exercises if their waiting period was particularly long, and in some cases generated momentum such that the user continued fetching until she had reached a particular learning goal.

Using Micro-Structures to Complement Micro-Waiting

Because expected time commitment may affect engagement more than actual time spent, learning exercises designed to capture user attention during waiting periods should be bite-sized and low-pressure. Users reported that a core benefit of WaitChatter was that it demanded very low time commitment, which ironically invited more frequent usage. In line with prior research on education [26], our research suggests that, beyond vocabulary flashcards, general learning tasks which impose a low cognitive load and can be reasonably learned in isolation may be more appropriate for wait-learning.

CONCLUSION

In this paper, we introduced the idea of wait-learning and evaluated a system that offers wait-learning opportunities in IM. We found that exercises displayed at the beginning of a potential waiting period receive higher engagement, and that exercises should be optional and demand a low cognitive load to minimize intrusiveness. While this work investigated the effectiveness of wait-learning within a single medium (IM), the general classes of measures that were investigated – the amount learned during wait time and the timing of learning opportunities – occur in other situations and generalize to other potential forms of wait-learning. In the future, it would be beneficial to investigate wait-learning in other media contexts and educational domains, enticing even the busiest users to engage in life-long learning.

ACKNOWLEDGMENTS

We thank Max Goldman and the members of the UID group at MIT CSAIL for all their help and feedback. This work was funded by Quanta Computer and MIT Lincoln Laboratory.

REFERENCES

1. Adamczyk, P. D., and Bailey, B. P. If not now, when?: the effects of interruption at different moments within task execution. In *CHI '04*, ACM (2004), 271–278.
2. Avrahami, D., Fussell, S. R., and Hudson, S. E. Im waiting: timing and responsiveness in semi-synchronous communication. In *CSCW '08*, ACM (2008), 285–294.
3. Avrahami, D., and Hudson, S. E. Communication characteristics of instant messaging: effects and predictions of interpersonal relationships. In *CSCW '06*, ACM (2006), 505–514.
4. Bailey, B. P., and Iqbal, S. T. Understanding changes in mental workload during execution of goal-directed tasks and its application for interruption management. *TOCHI 14*, 4 (2008), 21.
5. Beaudin, J. S., Intille, S. S., Tapia, E. M., Rockinson, R., and Morris, M. E. Context-sensitive microlearning of foreign language vocabulary on a mobile device. In *Ambient Intelligence*. Springer, 2007, 55–72.
6. Böhmer, M., Lander, C., Gehring, S., Brumby, D. P., and Krüger, A. Interrupted by a phone call: exploring designs for lowering the impact of call notifications for smartphone users. In *CHI '14*, ACM (2014), 3045–3054.
7. Cai, C. J., Guo, P. J., Glass, J., and Miller, R. C. Wait-learning: leveraging conversational dead time for second language education. In *CHI'14 Extended Abstracts*, ACM (2014), 2239–2244.
8. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research 12* (2011), 2493–2537.
9. Czerwinski, M., Cutrell, E., and Horvitz, E. Instant messaging: Effects of relevance and timing. In *People and computers XIV: Proceedings of HCI*, vol. 2, British Computer Society (2000), 71–76.
10. Dearman, D., and Truong, K. Evaluating the implicit acquisition of second language vocabulary using a live wallpaper. In *CHI '12*, ACM (2012), 1391–1400.
11. Dempster, F. N. Effects of variable encoding and spaced presentations on vocabulary learning. *Journal of Educational Psychology 79*, 2 (1987), 162.
12. Dornyei, Z., and Ottó, I. Motivation in action: A process model of l2 motivation. *Working Papers in Applied Linguistics* (1998), 43–69.
13. Ebbinghaus, H. *Memory: A contribution to experimental psychology*. No. 3. Teachers college, Columbia university, 1913.
14. Edge, D., Searle, E., Chiu, K., Zhao, J., and Landay, J. A. Micromandarin: mobile language learning in context. In *CHI '11*, ACM (2011), 3169–3178.
15. Fogarty, J., Hudson, S. E., Atkeson, C. G., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J. C., and Yang, J. Predicting human interruptibility with sensors. *TOCHI 12*, 1 (2005), 119–146.
16. Gassler, G., Hug, T., and Glahn, C. Integrated micro learning–an outline of the basic method and first results. *Interactive Computer Aided Learning 4* (2004).
17. Godwin-Jones, R. Emerging technologies from memory palaces to spacing algorithms: approaches to secondlanguage vocabulary learning. *Language, Learning & Technology 14*, 2 (2010).
18. Isaacs, E., Walendowski, A., Whittaker, S., Schiano, D. J., and Kamm, C. The character, functions, and styles of instant messaging in the workplace. In *CSCW '02*, ACM (2002), 11–20.
19. Isaacs, E., Yee, N., Schiano, D. J., Good, N., Ducheneaut, N., and Bellotti, V. Mobile microwaiting moments: The role of context in receptivity to content while on the go.
20. Kahneman, D. *Attention and effort*. Citeseer, 1973.
21. Ling, R., and Baron, N. S. Text messaging and im linguistic comparison of american college data. *Journal of Language and Social Psychology 26*, 3 (2007), 291–298.
22. Miyata, Y., and Norman, D. A. Psychological issues in support of multiple activities. *User centered system design: New perspectives on human-computer interaction* (1986), 265–284.
23. Nardi, B. A., Whittaker, S., and Bradner, E. Interaction and outeraction: instant messaging in action. In *CSCW '00*, ACM (2000), 79–88.
24. Ofek, E., Iqbal, S. T., and Strauss, K. Reducing disruption from subtle information delivery during a conversation: mode and bandwidth investigation. In *CHI '13*, ACM (2013), 3111–3120.
25. Pavlik, P. I., and Anderson, J. R. Using a model to compute the optimal schedule of practice. *Journal of Experimental Psychology: Applied 14*, 2 (2008), 101.
26. Sweller, J., Van Merrienboer, J. J., and Paas, F. G. Cognitive architecture and instructional design. *Educational psychology review 10*, 3 (1998), 251–296.
27. Takemae, Y., Chaki, S., Ohno, T., Yoda, I., and Ozawa, S. Analysis of human interruptibility in the home environment. In *CHI'07 Extended Abstracts*, ACM (2007), 2681–2686.
28. Tree, J. E. F., Mayer, S. A., and Betts, T. E. Grounding in instant messaging. *Journal of Educational Computing Research 45*, 4 (2011), 455–475.
29. Trusty, A., and Truong, K. N. Augmenting the web for second language vocabulary learning. In *CHI '11*, ACM (2011), 3179–3188.
30. Webb, S. The effects of repetition on vocabulary knowledge. *Applied Linguistics 28*, 1 (2007), 46–65.