Spaced Repetition and Anki: A Practical Introduction

Connor Harris

How to download

- Desktop client: apps.ankiweb.net/#download
- On iOS: "AnkiMobile Flashcards" (costs $25 but worth it)—beware imitators with similar names!
- Free third-party clients for Android
- Ankiweb offers free accounts for synchronizing between devices

Spaced repetition: the basic ideas

- Basic sales pitch: long-term memorization with **near-perfect reliability** and **orders of magnitude less effort**
- Based on well-replicated cognitive psychology findings:
  - **Testing effect:** quizzing yourself on knowledge > just reading it
  - **Spacing effect**: reviews spread over time > cramming
- Spaced repetition software (SRS) schedules reviews for you to keep thousands of items in memory with a few minutes of quizzes/day
- Most popular SRS: Anki
  - Extensively customizable with CSS and JavaScript (though a bit janky)
  - Good support for images, audio, mathematics (with LaTeX markup)
  - Desktop and mobile clients, synchronizable with a free web account

Anki is useful for **anything** that requires memory!

- Most popular uses: **med school** and **foreign languages**. Others:
  - **History:** "Treaty that ended World War I", "Who was elected president in 1976?"
  - **Geography:** "Capital of Finland", "Driving distance from Houston to Dallas"
  - **Computers:** "Python method for uppercasing a string", "HTTP status code for OK"
    - Less time searching API references = more time solving hard problems!
  - **Chemistry:** "Structure of azide ion", "Products of sodium metal/water reaction"
  - **Game rules and tactics:** "What does 'roll at disadvantage' mean?", "Most common Black responses to Queen's Gambit"
  - **Poems and plays:** "Fill in the gap: 'O Romeo, Romeo, […] would smell as sweet'"
  - **Job knowledge:** "How do I get the office printer to print double-sided?"
  - **Interpersonal:** "Name the friend in this photograph"; "Jane's husband's birthday"

"Big concept" subjects

- Some subjects are lots of small facts (e.g. history, geography, biology, programming, foreign languages) and some are a few big, hierarchical concepts with lots of implications (e.g. physics, mathematics)
- Anki (and brute memorization more generally) is more effective with small facts than with big concepts
- But memorization can help with parts of big-concept subjects: definitions, formulas (e.g. important trig identities and integrals), important steps/key ideas in proofs

Basic Anki review workflow

- Open a deck of flashcards and look at the front side of the first card scheduled for that day's review
- Tap/click/press enter to see the answer
- Tell Anki if you got it right
  - If no: you'll see it again in a few minutes, then frequently for the next several days
  - If yes: Anki will schedule another review farther into the future
- Repeat until you're done for the day (cards you don't get to are backlogged for the next day)
- No need to do it all in one sitting—you can do ten cards in line at the supermarket instead of checking Twitter

Notes, types, fields, decks

- Basic unit of knowledge: **note**, with a **type** that specifies:
  - **Fields**: what information each note contains
  - **Templates** that create cards (= quiz questions) from each note
- Anki has a few basic note types built in
- You combine notes into **decks** for a particular subject
  - Decks can have multiple note types
  - Most scheduling algorithm parameters are set per deck
  - You can tell the scheduler not to give you multiple cards from the same note on the same day ("Bury related cards"; "bury" = "postpone one day")

Example note type

- To learn Spanish vocabulary, you might have a note type with fields **Spanish**, **English**, and **Gender** (only for nouns). Sample notes:
  - **Note 1:** Spanish: "bosque", English: "forest", Gender : "masc."
  - **Note 2:** Spanish: "ciudad", English: "city", Gender: "fem."
  - **Note 3:** Spanish: "correr", English: "to run", Gender: [blank]
- Template 1: quizzes you on English word
  - Front: {{Spanish}}
  - Back: "{{FrontSide}} <hr/> {{English}}
  - Resulting card for note 1: "*bosque*" →"forest"
    - (FrontSide is a pseudo-field that lets you keep front-side information on the screen while viewing the back side)

Example note type cont'd

- Template 2: quizzes you on gender
  - Front side: {{#Gender}}Gender of {{Spanish}}{{/Gender}}
  - Back side: {{FrontSide}} <hr/> {{Gender}}
- Sample resulting card: "Gender of *ciudad*" → "fem."
- {{#Gender}}...{{/Gender}} is conditional markup: contents rendered only if Gender is not blank
  - Anki won't create cards with blank fronts, so this template is skipped for notes with a blank Gender field, e.g. verbs such as *correr* (you might also want to skip this card if gender is obvious from the word ending, e.g. *-ción* = always feminine)
- A real deck might use multiple note types: "Noun" (with a gender field), "Verb" (no gender, but fields for irregular forms), etc.

Deck best practice: don't mix subjects

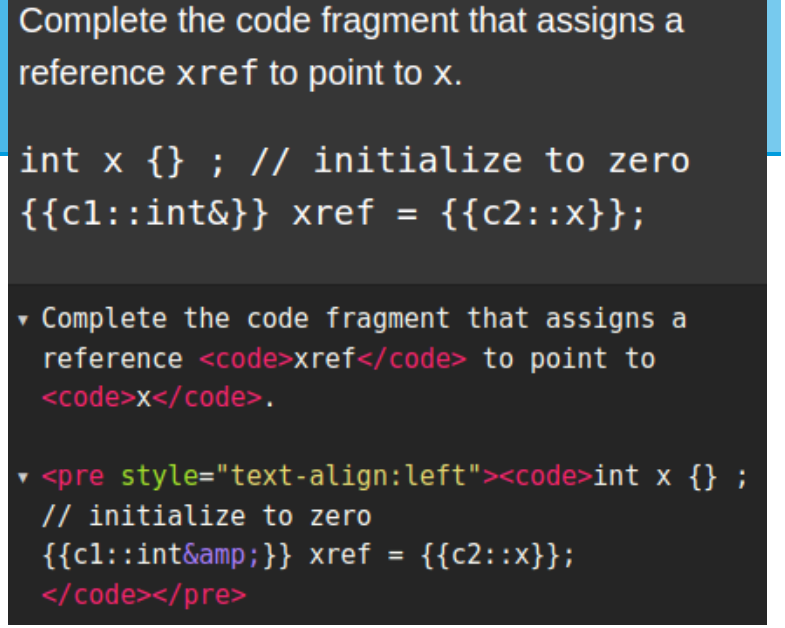
- Scheduling parameters are set per deck. For harder subjects, you may want fewer new cards/day and shorter review intervals
- You review one deck at a time, so the only context for a card besides its content is which deck it came from. So mixed-subject decks mean:
  - More mental gear changes
  - More context needed on cards (= more reading = slower reviews), e.g. "Method to append to a list **in C++**"/"Method to append to a list **in Python**". With separate Java and Python decks, cards can be shorter.
- You can organize decks hierarchically (e.g. "Programming::C++" and "Programming::Python") and choose between reviewing a subdeck or a mix of all subdecks in a superdeck

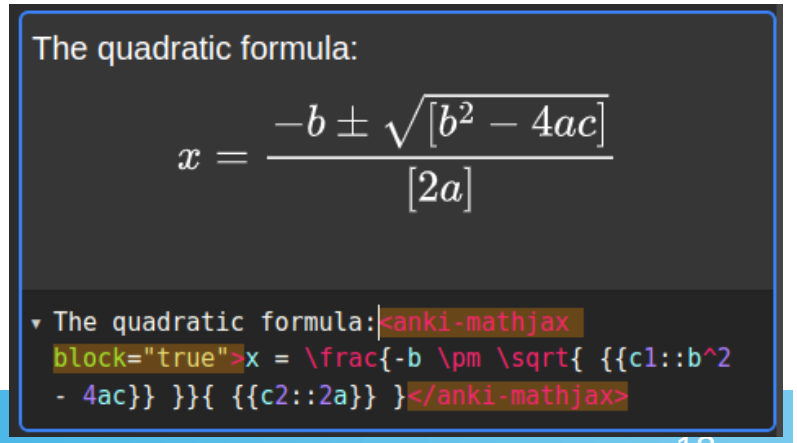
Cloze deletion

- Special fill-in-the-blank note type. Useful for grammar drills (e.g. "fill in the right form of the verb") and much more
- Efficient way to create multiple cards from one fact
- Syntax: deletions in double braces with a card number + optional hint
  - "The {{c1::Sabine}} River divides {{c2::Texas::state}} and {{c2::Louisiana::state}}" →
    - Card 1: "The **[…]** River divides Texas and Lousiana."
    - Card 2: "The Sabine River divides **[state]** and **[state]**."
  - Hints are useful when a blank could be filled multiple ways: "Frobnication increases transmogrifier output by **[how much?]**" versus "... by **[what mechanism?]**"
  - Reusing a number produces multiple blanks on one card, useful if the content of one blank gives too big a clue about another

Other uses of cloze

- Cloze deletions in code snippets to learn programming language syntax (pictured: the card I made to learn C++ reference syntax)
  - Double-colon is valid syntax in some languages; insert ‍ in HTML editor to stop Anki from reading it as part of the tag
- Learning a poem or monologue? Make each phrase its own cloze deletion card with a few words on each end for context
- You can delete parts of math formulas





Basics of the scheduling algorithm

- Every card has an **ease**
  - Starting ease assigned and configurable per deck; by default 2.5
  - Wrong answers (or correct answers marked "Difficult") decrease ease by a configurable increment; answers marked "Easy" increase ease
- Ease = review interval multiplier: if you saw a card with ease 2.5 after 10 days and marked it correct (with default "Good" response), you'll see it again in 25 days
  - "Difficult" and "Easy" answers will change the scheduled time for the next review as well as altering ease
- Review intervals for easy cards grow exponentially (to > 1 year after 7-8 reviews), focusing reviews on a handful of tough cards

Minimum information principle

- Keep cards short! Aim for <2 seconds and ~0 thought per review
  - Long cards that take more thought = slow, tiring review sessions
  - You likely won't remember extra information if you're not quizzed on it
  - Long cards are sometimes unavoidable (e.g. theorems with complicated premises), but avoid when you can
- Answers should be short enough to be clearly right or wrong—don't let yourself be tempted to mark half-right answers as right
- Keep the text on cloze cards short (or divide into shorter Q/As)
- More: https://www.supermemo.com/en/blog/twenty-rules-of-formulating-knowledge

Minimum information principle con't'd

- Question with long, unmemorable answer:
  - Who was Edith Wharton? → American novelist (1862-1937) who criticized the superficiality and hedonism of New York high society, which she was born into but abandoned by moving to France to escape the scandal of her divorce in 1913. Her best regarded novel, *The Age of Innocence*, won the 1921 Pulitzer Prize.
- Better: break into multiple cards:
  - When was Edith Wharton born? → 1862
  - Why did EW move to France? → To escape a divorce scandal
  - Main setting of EW's novels → New York high society
  - How did EW portray high society? → Superficial and hedonistic
  - Novel that won EW the Pulitzer Prize → *The Age of Innocence*

Minimum information goes for clozes, too

- Example: "A {{c1::buddy allocator}} is a {{c2::segregated storage}} memory allocator in which block sizes are all {{c3::powers of two}}. It makes smaller blocks by {{c4::splitting larger blocks in half}}."
- Text takes a bit too long to read. We can split into simple Q/A cards:
  - What memory allocator makes small blocks by splitting large blocks in half? → Buddy allocator
  - How does a buddy allocator create small blocks? → Splitting large blocks in half
  - What type of allocator is a buddy allocator? → Segregated storage
  - What block sizes can a buddy allocator create? → Powers of two
- Tradeoff: multiple notes are harder to create/edit than one cloze (but you'll spend more time reviewing a note than making it)

Target both directions of an association

- In foreign languages, "passive" vocabulary (words you recognize) is bigger than "active" vocabulary (words you can produce)
- Similar phenomena in other domains
  - E.g. "What does arr.unshift(x) do?" → "adds x to the front of arr" likely easier than "how do you add x to the front of arr?" → "arr.unshift(x)"
  - "When was Jimmy Carter elected president?" → "1976" versus "Who was elected president in 1976?" → "Jimmy Carter" (the first is a fact about Carter, the second about 1976; depending on prior knowledge, one might be easier than the other)
- Be sure to cover all directions!
  - Use default bidirectional note type or cloze: "The {{c1::.unshift()}} method {{c2::appends to}} the {{c3::front}} of an array", "{{c1::Jimmy Carter}} was elected president in {{c2::1976}}."

Advice: use premade decks with caution

- AnkiWeb has user-contributed **premade decks** on many subjects (esp. foreign languages)
- These might seem tempting, but general advice is **not to use them**. Why?
  - Quality is variable
  - Condensing knowledge into note form yourself will help you remember it
  - You might see cards before you've learned their context, and memorizing word strings by brute force is harder than fitting facts into an existing cognitive framework
- Exception: decks made to fit a specific textbook or standardized curriculum could be useful if you're following that textbook

Advice: avoid long list answers

- Lists are hard to remember as a single unit. Bad question:
  - "Who were the members of the Beatles?" → "John Lennon, Paul McCartney, George Harrison, and Ringo Starr"
  - (For the sake of this slide, Pete Best does not exist)
- Better: cloze deletion; targets one item at a time
  - "The Beatles were {{c1::John Lennon}}, {{c2::Paul McCartney}}, {{c3::George Harrison}}, and {{c4::Ringo Starr}}"
- Best: encode the knowledge in questions with shorter answers
  - "Who was the Beatles' drummer?" → "Ringo Starr"
  - "Who wrote most of the Beatles' songs?" → "Lennon and McCartney"

Advice: arbitrary categories

- Hard to remember. One tip: make a color-coding scheme (possible in Anki with custom JavaScript)
- Things you may want to color-code:
  - Languages: noun gender, Russian accent pattern, Chinese tone (color is more memorable than diacritics)
  - Programming: methods that modify vs. copy their arguments
  - History: politicians' parties or factions
  - DnD: what kind of check (STR, CON, INT, etc.) an action requires
- (My friend Campbell Nilsen is building an SRS platform for classroom use with more ergonomic features (incl. color coding): www.gofundme.com/f/nextgeneration-spaced-repetition-software)

Two common causes of difficult cards

- **Interference:** mental crossed wires—two prompts are similar enough that you can't remember which answer goes to which prompt
  - Common with similar-sounding words, e.g. two difficult words in my Spanish deck: *quijada* (jaw) and *quebrada* (crack, ravine)
  - With programming languages, common to mix up minor differences in syntax between languages
- **Subchunk failure:** trying to memorize a large structure when you don't have the smaller chunks memorized
  - E.g. it's easier to memorize a molecule as three functional groups whose names you already know rather than fifteen atoms

Subchunk failure: use smaller chunks

- Suppose you need to learn the quadratic formula. You'll have trouble remembering the whole formula at first, so break it into chunks on separate notes:
  - Cloze-delete terms of the discriminant: "The discriminant of a quadratic is {{c1::b^2}} – {{c2::4ac}}."
  - Produce the entire discriminant of a quadratic.
  - Produce the entire quadratic formula. Now that "discriminant" is a single concept in your memory, this should be easier.
- Also works in other contexts; e.g. cloze-delete individual syllables of difficult words or functional groups of chemical structures.

Dealing with interference

- Suspend one interfering card until you're solid on the other. The simplest and often the best choice.
- Target the error with partial-word cloze deletions
  - E.g. I keep confusing Spanish *quijada* (jaw) and *quebrada* (crack, ravine). I can target this with "Spanish **qu[...]ada** = jaw".
- Invent a mnemonic and include it as extra information on the back—AI chatbots are often good at suggesting mnemonics