Preliminaries

Data we provide: news.txt (a collection of news articles), on the homepage.
Whenever you are asked a question that requires you to implement an FST, we expect you to write an FST or a script that generates an FST to solve that question. Please send your answer as an ipython notebook (.ipynb).

Special characters:

Within OpenFST:

- space becomes <space>
- newline becomes <newline>
- An epsilon transition is <epsilon>
- Your tokenizer should indicate the beginning of each token with <t> and the end of each token with < /t>

You can assume that all data will contain ASCII characters: (in decimal range 32-127). This excludes tabs and return feeds of all sorts. Each line will always end with <newline>.

Exercise 1

Build a tokenizer that makes every character its own token.

1. Visualize this automaton include the result as your answer.
2. How many states, arcs, and accepting states does this automaton have?
3. Use “determinize” to modify your previous non-deterministic FST to make a deterministic FST.
4. How many states, arcs, and accepting states, does this FSA have?
Exercise 2

Build a tokenizer that splits tokens on spaces and newline characters only.

1. How many states, arcs and accepting states does this FSA have?
2. Visualize this automaton include the result as your answer.

Exercise 3

Modify the above space-delimiting tokenizer to separate all periods as a separate token.

1. How many states, arcs and accepting states does this FSA have?
2. Visualize this automaton include the result as your answer.
3. Explain how this tokenizer operates.

Exercise 4

Build a transducer that acts as a trivial stemmer, keeping only the first 3 characters of each token and discarding the rest of the word.

1. Explain how this stemmer operates.
2. How many states, arcs, and accepting states, does this FSA have?

Exercise 5

Programmieren Sie einen Transduktor in OpenFST, so dass er jede beliebige Zahl (\(\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}\)^*) als Eingabe nimmt, und ihre kanonische Repräsentation als Ausgabe, d.h. dieselbe Zahl, aber ohne initiale 0: aus 0220 wird 220 etc.