A *regular language* is a set of words over an alphabet Σ that can be described completely by a regular expression, i.e., using letters from Σ, the empty word ϵ, and applications of concatenation, alternation (union), and Kleene star. A *deterministic finite automaton* (DFA) is a quintuple $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$, where Q is a set of states, $q_0 \in Q$ is the initial state, $F \subseteq Q$ is the set of accepting states, and $\delta : Q \times \Sigma \to Q$ is a transition function, extended to $\delta : Q \times \Sigma^* \to Q$ in the usual recursive way. We say that $\mathcal{D}$ *accepts* a word $w \in \Sigma^*$ if and only if $\delta(q_0, w) \in F$. The class of regular languages is exactly the class of languages accepted by DFAs.

Given a regular language there is a DFA with a minimal number of states that accepts it, called its *minimal DFA*, and is unique up to isomorphism; moreover, minimality depends only on intrinsic properties of the DFA itself. Thus we may study the "complexity" of a regular language through its minimal DFA. One such measure is *state complexity*, the number of states of the minimal DFA.

The focus of my research has been another measure of complexity, known as *syntactic complexity*. For a given DFA, any non-empty word $w \in \Sigma^*$ induces a transformation t on the set of states given by $t(q) = \delta(q, w)$. The set of all possible transformations in the DFA is a semigroup under composition, and called the *transformation semigroup* S of $\mathcal{D}$. The syntactic complexity of a regular language L can be defined to be the size of the transformation semigroup of its minimal DFA. Clearly, the maximum number of transformations on n states is n^n; however, by restricting study to subclasses of regular languages, tighter bounds can be found. My research focused on the finite/cofinite, definite, and reverse definite subclasses, where cofinite languages are the complements of finite languages, and definite (reverse definite) languages have the form $E \cup \Sigma^* F$ ($E \cup F\Sigma^*$) for finite languages E and F.

To find syntactic complexity upper bounds we applied the same technique throughout: characterize the allowable transformations for the minimal DFAs of the class, and then find the size of the largest semigroup containing only these transformations using combinatorial methods. For finite languages, there is a numbering on the states of the minimal DFA such that for any word $w \in \Sigma^*$, $\delta(i, w) \geq i$ with equality if and only if $i = n$. The number of such transformations satisfying this property is $(n-1)!$; moreover, the set of all such transformations is in fact a semigroup, and hence this provides a tight upper bound on the syntactic complexity. A similar correspondence can be found for reverse definite languages, with only a slight modification: $\delta(i, w) \geq i$ with equality if and only if $i \in \{n-1, n\}$. Again, the set of all transformations with this property is a semigroup of size $(n-1)!$.

For definite languages, transformations must be *non-permutational*, where a transformation is permutational if there is a subset $X \subseteq Q$, $|X| > 1$ such that t restricted to X is a permutation of X. There are n^{n-1} non-permutational transformations on n states. Unlike the previous cases, however, the set of all non-permutational transformations is *not* a semigroup. We found a maximal non-permutational semigroup of size $\lfloor e \cdot (n-1)! \rfloor$, hence providing a *lower* bound on the maximum syntactic complexity of definite languages. We conjecture this is a tight *upper* bound as well.

We have also studied the minimal number of generators required to reach these bounds. This corresponds to the alphabet size required, as the transformation semigroup of a DFA is generated by the transformations corresponding to single-letter inputs. For the finite/cofinite and reverse definite cases, at least $(n-1)! - (n-2)!$ generators are required to reach the upper syntatic complexity bound of $(n-1)!$. For the semigroup we found for definite languages, a minimum of $\lfloor e \cdot (n-1)! \rfloor - \lfloor e \cdot (n-2)! \rfloor$ generators are required.

We have verified the conjecture for definite languages up to a state complexity $n \leq 4$ through computational enumeration, but have yet to prove this conjecture. The technique we used may be applicable to more complex classes of regular languages, notably the generalized definite and locally testable languages, which lie directly above definite/reverse definite languages in the dot-depth hierarchy of star-free languages.