Methodology: We performed some preprocessing (loading files into memory, filtering sentences with a stoplist, etc.), then started the timer and computed the optimal solutions for all the files in a given run. We first did three “dummy” runs to warm-up the JVM, then averaged the runtime of the next ten runs. On the CNN/DailyMail database, we tested various numbers of files -- as expected, the runtime is linear in the number of files processed. The differences in ROUGE scores are due to the possibility of multiple optimal solutions, and the fact that optimality doesn’t directly correspond to ROUGE. We tested three ILP solvers, and found that GLPK was more than twice as fast as the competition on these problems. Gurobi and lp_solve weren’t recorded for the more challenges instances, as preliminary tests showed that the same patterns held. These tests are for the unweighted ILP solution to MCP:

Data:
The CNN/DailyMail articles were relatively short. As such, we constrained our solver to three sentence summaries (the average length of the reference summaries was ~3.75 sentences).

Since we established that the runtime in linear in the number of files (assuming similar file size), we used a smaller set of much larger files to see how the ILP runtime increases with file size. We picked the 100 largest files (minus two outliers) for one data set and tested the two outliers separately. Since the files are bigger, we allowed our solver to see how many sentences the target summary contained and match that length. The increased file size combined with the increased summary length increased the runtime significantly.

The table, with runtimes in milliseconds and ROUGE scores in parentheses.

	lp_solve	Gurobi	GLPK
CNN: 1k	12002 (0.43913)	10260 (0.43817)	4775 (0.43900)
CNN: 5k	58896 (0.44000)	52796 (0.43909)	26389 (0.43933)
CNN: 10k	132760 (0.43980)	89762 (0.43987)	52483 (0.43959)
Legal Cases Biggest 100:	1328445 (0.60113)	----	359815 (0.60357)

Our last “challenge” instance was obtained through Project Gutenberg, “a volunteer effort to digitize cultural works”. We concatenated several of the top 100 books: Moby Dick, Pride and Prejudice, The Substitute Millionaire, Heart of Darkness, A Tale of Two Cities, The Adventures of Sherlock Holmes, and Treasure Island into a large text file with 30,000 sentences. We don’t have a reference summary here: we’re just encoding the file and testing the solvers. We also have two legal cases that were much larger than the other cases.

The ILP solver struggled to create reasonably long summaries for these instances (we set a timeout of roughly an hour). We tried to see how long of a summary we could generate while remaining within a reasonable time. Interestingly enough, it seemed like there was a single

sentence between taking less than a minute and not finishing within an hour. Perhaps some of the sentences are much more valuable than the rest, and once those are taken the problem becomes much more challenging.

	K = 3	K = 6	K = 9	K = 10	K = 13	K = 14
Legal Case: 08_498 (1.3 MB, ~10k sentences)	7379 (0.090638)	8586 (0.126808)	12756 (0.183829)	> 1 hour (???)	----	----
Legal Case: 07_1062 (2.5 MB, ~19k sentences)	20199 (0.21339)	20000 (0.29485)	21868 (0.37431)	28388 (0.38880)	25311 (0.41879)	> 1 hour (???)
English Texts (3.7 MB, ~34k sentences)	118505	121757	126085	130884	138035	> 1 hour (???)