Terms of Service Summarization

Mitchell Falkow
RPI
falkom@rpi.edu

Ji hann Hong
RPI
jnh2016@gmail.com

Andrew Ma
RPI
maa5@rpi.edu

Shoshana Malfatto
RPI
malfas@rpi.edu

Chris Tang
RPI
tangc6@rpi.edu

Abstract

Nobody wants to–or has the time to–spend reading the Terms of Service (ToS) agreement of a product before clicking *Accept*. As a result, the average individual haphazardly agrees, giving little thought as to what lies in the countless lines of legal text they now are legally obliged to submit to. Modern companies typically hide terms within the fine print of these documents, relinquishing users from the right to sue, or detailing the process by which said companies will sell users' data. Meanwhile, the average user will remain in total ignorance.

For this reason, our team worked to condense, summarize, and highlight the rules, warnings, and rights inside user agreements to give users more concise overviews of the terms that they agreed to. This problem was tackled with a two-pronged unsupervised and supervised approach. The supervised models employed were k-NN, linear perceptron, and SVM. In the end, our 1-nearest neighbor model produced the best results out of the three supervised approaches, for reasons we will discuss in our paper.

1 Background and Motivation

Legal documents, such as Terms of Service, Terms and Conditions, and End-User License Agreements[1] are infamously hard to read. Most users will often skip over the entire document and click the *I agree* button without the slightest hesitation. When people do this, it gives companies the ability to hide less agreeable conditions, like making the user to relinquish the right to sue or ownership of their data to the company. If a user could see these terms outright, then it might make them change their mind about using a service.

[1]For simplicity's sake, we will refer these documents collectively as *ToS* agreements

Our goal is to create a system that can bring attention to any suspicious or notable statements relating to payments, account termination, intellectual property, legal action, data privacy, harassment policies, or any of the other topics that could appear on a ToS document. Our hope is that our research could be used to perhaps let users know exactly what they are agreeing to in a ToS documents.

In terms of related examples, we found very few. There exist resources where a human does the extraction, like tl;drLegal (tld), which summarizes software licenses, and Terms of Service; Didn't Read (tos) which summarizes the good and bad qualities of many major websites' Terms of Service documents.

Much of the other examples on the topic of legal summarization as it relates to web-services focus heavily on company privacy policies. While related, these two document types serve very different purposes. Privacy policies often focus purely on the usage of user data, meanwhile ToS documents list rules to which a user must agree if they want to use a service. ToS documents can also include conditions both related and unrelated to privacy, but generally serve to protect the organization from legal allegations. Terms of Service often include disclaimers about the service. The key difference between the two documents is that privacy policies often do not require the user's approval, but ToS documents do.

2 System Architecture and Approach

Our approach consists of two separate systems:

1. System 1 (see Figure 1) expands upon LexRank's algorithm with a pre-processing and post-processing architecture in order to enhance readability.
2. System 2 is an supervised machine-learning

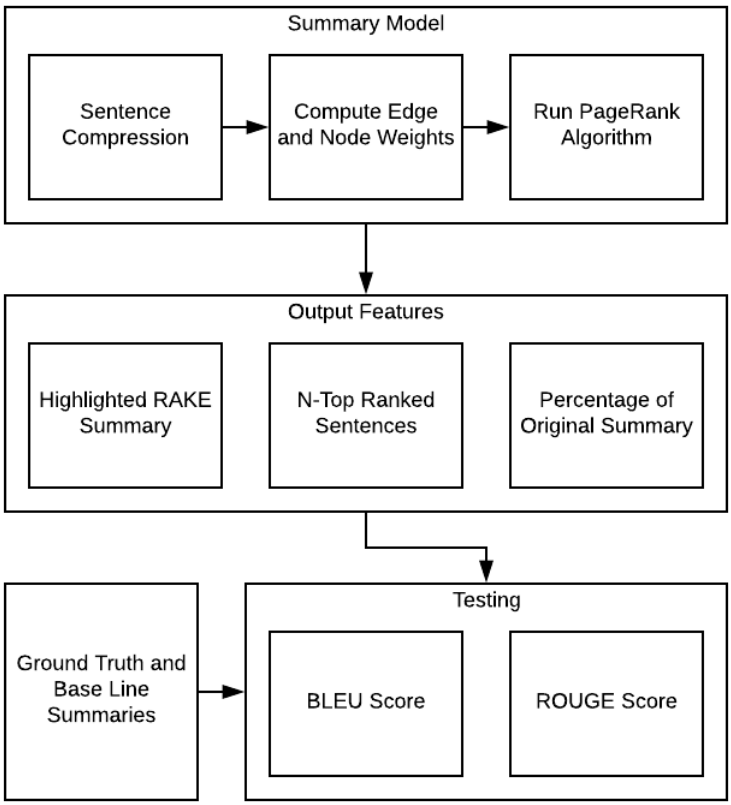

Figure 1: System 1 Architecture (Unsupervised, LexRank-based)

system based off eight features we generate from a document. This system treats the task of summarization as a classification problem, and can utilize several classification models to generate a summary. This system uses the 1-NN machine learning model.

Both System 1 and System 2, and their respective outputs, are independent of each other.

2.1 Base Input

Both systems are provided a ToS document in plain-text format. The text is sentence segmented, and in some cases, receives an additional layer of preprocessing.

2.2 System 1 – Unsupervised, LexRank-based system

The LexRank framework relies heavily on the data structure representing the sentences. We utilized a matrix, as the data is dense and strongly connected. The data model can store Terms of Service documents, with information on the document-level, sentence-level, and word-level. The implementation is based on LexRank (Erkan and Radev, 2011) paper and pre-existing implementations, specifically the sumy library (Belica). We are able to use the power method algorithm for computing stationary distribution in Markov chains, as described in the LexRank paper, to determine the ranking of the sentences.

We wanted to extend the algorithm around LexRank to work especially well for ToS documents. To achieve this goal, we decided to do text normalization specific to our task, followed by sentence compression as a preprocessing step. The output from sentence compression is then pipelined into the LexRank methods. We then try different ways of displaying the output so that the summary is more informative.

2.2.1 Input

This system supports legal documents stored as a text file(s). Following **Figure 2**, the data that will be used will come from the current websites displaying the ToS of a company. This input will be normalized. From the input, the core module will parse and store the document in our matrix for the base LexRank algorithm.

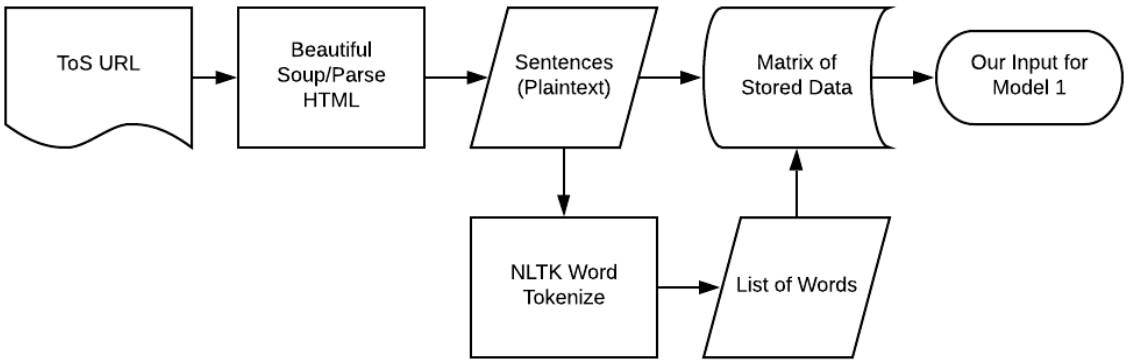

Figure 2: System 1 Input Pre-Processing

2.2.2 Text Normalization

We wanted to have the option to grab a ToS document using a URL, which required parsing HTML. Having the element tags allows us to deal with special cases, like the "Code of Conduct" sections which generally start out as a list saying that the reader must agree to *not* do the following actions, under no circumstances, followed by a list of actions. Parsing just the text from a ToS does not show that these sentences are in a list, and might include them in a summary as an action that you *should* do. Having the list element tags, we can prepend the negating sentence to all of the items in the list.

2.2.3 Sentence Compression

Ideally, sentence compression makes the ToS summary easier to read, and prevents longer sentences from being ranked higher than shorter sentences because they have more words. For our implementation, we used the parse-and-trim approach (Xu and Grishman, 2009) which was based on Chinese syntax, and the older Hedge Trimmer method (Dorr et al., 2003) which is cited in the first paper and generates headlines for newspaper articles in English using the parse-and-trim approach.

We converted sentences in the input ToS into syntactically parsed trees using the NLTK interface for the Stanford Parser (Manning et al., 2014). Branches of the tree are iteratively removed based on rules about the importance of certain syntactic structures in a sentence and the significance of the phrase or clause, until the sentence is deemed short enough. The word significance score, as described in (Xu and Grishman, 2009) was

$$tf \times idf$$

The rules were based on ideas from both papers. Adverb phrases, parenthetical elements, fragments, and interjections are removed unless they are deemed too important. Then outer simple declarative clauses, noun phrases, and verb phrases are removed iteratively. The next iterative step is to remove the rightmost trailing prepositional phrases. After trimming the branches, the tree is converted back to a sentence.

There are a few parameters that can be passed to the SentenceCompression class, including ω, which adds weight to proper nouns and β, which sets the preferred maximum number of characters per sentence.

In most cases these sentences make sense grammatically because the trimming is based on syntactic structure. The output was compared with sentence compression test data from Google (Datasets, 2017).

2.2.4 Output and Format Features

The core output feature will be an extractive summary at default granularity.

Another feature that is planned is an adjustable granularity of the extractive summary. This is inherently achievable through the data structure. Given the ranking of the sentences and original sentence order, the summary size can be adjusted by adding more sentences or removing sentences while still maintaining logical order.

The last main feature we hope to implement would be a summary highlighting tool. By highlighting, the extractive sentences mapped to the full ToS would help a reader disseminate the sentences importance in context. Another layer of highlighting emphasis can be on the word/phrase

level. Given signature term extraction, the highest rated phrases can be highlighted in a sentence.

2.3 System 2 – Supervised Learning

This system was inspired by research from (Goldstein et al., 1999). The system treats the summarization task as a classification problem, whereby a classifier determines whether or not a data point–in this case, a data point is a single sentence–should be in one of two classes: (-1) not in the summary, or (1) in the summary. The decision to include a sentence is based on a list of weighted features. We then utilize an array of supervised machine learning approaches to create an extractive summary of the legal text. The models implemented and tested were:

- Perceptron linear classifier (Rosenblatt, 1958)
- k-Nearest Neighbors (k-NN) model
- Support vector machine (Boser et al., 1992)

Each of the models was implemented using the *Sci-Kit Learn* package for Python (Pedregosa et al., 2011).

2.3.1 Input and Training Data

This system takes in sentence-segmented ToS documents. Each sentence within the text is viewed as a single data point for all implementations of each model.

For training, the system must be given the segmented ToS document and their corresponding ground truth summaries.

Both the input and the training data is sent through a pre–processing pipeline. Documents are parsed by a script which separates each sentence onto its own line, with markers denoting where paragraphs started (for use in feature calculations, this will be discussed later). The text output is then fed into each of the three supervised models for both training and testing purposes.

2.3.2 Features

We used the following 8 features to train with:

- Length of sentence ignoring stop words
- Number of commas in the sentence
- Order of the sentence from the top of document divided by total number of sentences in the document to normalize it between 0 and 1
- Order of the sentence from the top of the paragraph it's in divided by total number of sentences in the the paragraph it's in to normalize it between 0 and 1
- Number of words in the sentence that are capitalized
- Number of words in the sentence that are in all capitals
- The current paragraph it's in divided by total number of paragraphs to normalize it between 0 and 1
- tf-idf score of the sentence

Most of the features are self explanatory. We will go into the tf-idf score of the sentence and how we calculated it. First we calculate the tf and idfs scores of each word in the document. Then we get the average tf scores for each sentence by looking for each non-stop word a sentence has and then averaging their respective tf scores. We do the same thing but with the idf scores this time. Once we have the tf and idf scores for the sentences, we combine to get the tf-idf score of the sentence.

2.3.3 Machine Learning Step

The system is given training data, which consists of ToS documents, the features, and a corresponding ground truth. The system uses the training data to construct the classifiers–perceptron, k-NN, and SVM. We selected 1-NN to the system's primarily machine learning model because it performed the best on the data given.

After training, given any input sentence and its features, the classifiers will assign a label $y \in \{-1, 1\}$ to the sentence.

2.3.4 Output

The system generates summaries from a given ToS document by classifying each sentence using the trained classifiers. Sentences which receive a label of $y = 1$ are included in the output, whereas sentences labeled $y = -1$ are excluded.

3 Datasets

At the start of this project we noticed that there was a large caveat to performing accurate analysis of results from each of the two systems–we could not find a comprehensive dataset. Any and all datasets would have required to be generated manually, unrelated, or paid for collectively. Since we lacked funding as a group, we opted to construct two datasets in the respective manners:

1. Hand-written and annotated summaries of a small set of ToS documents.

2. The aptly named *SMMRY-ToS;dr* corpus. Using the ToS;dr as a reference, we obtained the URLs of approximately 200 ToS and privacy policy documents. We then used the python package BeautifulSoup (Richardson) to parse hand-retrieved HTML files in addition to scraping the links individually. The scraped text was run through a pre-constructed summarization API, SMMRY.com (smm). Summaries that were generated using SMMRY.com were capped at 7 sentences long. Results from this process were stored and used as corpus.[2]

4 Final Results

4.1 System 1 vs. Ground Truth

Here are our results compared to our ground truths. Each model summary and its ground truth counter were constrained to be the same size.

Company	Doc. Type	ROUGE
Amazon Alexa	ToS	43.74%
Atlassian	ToS	26.90%
Instagram	ToS	36.62%
Reddit	ToS	32.98%
Rovio	ToS	36.57%
Twitter	ToS	30.01%

Table 1: System 1 ROUGE-1 Scores Against Chris's Ground Truths

4.2 System 1 vs. SMMRY.com

Here are our results compared to summaries produced by SMMRY.com

[2]The massive lack of training and testing data for System 2 was a major problem. As such, we were were forced to create this second corpus. We recognize the flaws in constructing a corpus based on another model's output, but due to time constraints, imperfect legal knowledge, and the lack of a comprehensive data set for a supervised model, we felt that some data, albeit flawed, was better than no data at all– provided that we recognize this in our analysis and conclusions. We also felt it would be interesting to see how our models compared to an already existing model.

Company	Doc. Type	ROUGE
Disqus	ToS	40.15%
Steam	ToS	49.14%
Google	Privacy	50.33%
Apple	Privacy	44.51%
Apple	Website ToS	44.15%
Vimeo	Privacy	54.61%
Bing	ToS	50.36%
Reddit	Privacy	46.20%
Disqus	Privacy	43.50%
Academia	ToS	26.98%
Apple	Membership	19.38%
Oath	ToS	47.27%
GoDaddy	Trademark	44.54%
iCloud	Privacy	45.89%
Softpedia	Privacy	41.66%
Groupon	Privacy	43.86%
ResearchGate	Privacy	51.82%
GoDaddy	Domains ToS	44.09%
Oath	Privacy	47.27%
IBM	Privacy	44.05%
ResearchGate	IP	43.62%
Vimeo	ToS	46.91%
500px	Privacy	46.14%
Flattr	ToS	50.28%
GoDaddy	Civil Law	41.87%
Google	ToS	50.34%
Twitter	ToS	59.85%

Table 2: System 1 ROUGE F-Scores Against SMMRY.com

4.2.1 System 1 vs. TOS;DR

A summary based on each ToS legal matter mentioned in TOS;DR. The example below specifically uses the Twitter ToS (twi). These sentences are extractive which preserves their length. Ellipses are used for report brevity in which the Analysis section will describe the important aspects each summary. Another constraint of this comparison was keeping each summary at 6 and 7 sentences.

TOS;DR/System 1 Rouge Score for Twitter TOS: 39.04%

Manual TOS;DR summary of Twitter

- By submitting, posting or displaying Content on or through the Services, you grant us...
- The changes will not be retroactive, and the most current version of the Terms...
- These Terms of Service (Terms) govern your access to and use of our services...

- In any case, you must be at least 13 years old, or in the case of Periscope 16 years old, to use the Services.
- You are responsible for safeguarding your account, so use a strong password and limit its use to this account.
- We may revise these Terms from time to time.

System 1 summary

- These Terms are an agreement and Twitter, Inc., 1355 Market Street, Suite 900, San Francisco, CA 94103 U.S.A..
- We reserve the right to access, read, preserve, and disclose...
- We may suspend or terminate your account or cease providing...
- You agree that this license includes the right for Twitter to provide, promote, and improve the Services and to make Content...
- If you want to reproduce, modify, create derivative works... you must use the interfaces and instructions we provide...
- You may not do any of the following while accessing or using...
- All disputes related to these Terms or the Services will...

4.2.2 Rovio Sentence Compression

Below is an example of sentence compression in the Rovio Terms of Service, and syntactically parsed tree before and after. The trimmed tree and sentence still make sense grammatically.

ORIGINAL sentence from Rovio ToS

```
(ROOT
(S
(NP
    (NP (DT The) (NN transfer))
    (PP (IN of) (NP (NNP Virtual) (NNS Items))))
(VP (VBZ is)
(VP (VBN prohibited)
    (PP (IN except)
        (SBAR
        (WHADVP (WRB where))
        (S
        (NP (NNP expressly))
        (VP
            (VBD authorized)
            (PP (IN in) (NP (DT the) (NNPS Services)))
            (, ,)
            (ADJP (RB as) (JJ applicable))))))))
    (. .)))
```

The original sentence: *"The transfer of Virtual Items is prohibited except where expressly authorized in the Services, as applicable."*

TRIMMED

```
(ROOT
    (S
    (NP
        (NP (DT The) (NN transfer))
    (PP (IN of) (NP (NNP Virtual) (NNS Items))))
    (VP (VBZ is) (VP (VBN prohibited) None))
    (. .)))
```

The trimmed sentence: *"The transfer of Virtual Items is prohibited."*

4.2.3 Twitter RAKE

The original RAKE summary highlighted high scoring phrases and words within a summary.

In order to represent the output of using RAKE in a summary, 4 word lists were created to express RAKE's ranking of key phrases. These sets were sorted from most important to least important.

RAKE's rankings greater than a score of 4: [3] articles/15789#specific-violations, non-assignable, tv/customer/portal/articles/2460220, appealing violations, /overview/terms/policy, /forms/dmca email, super hearts, /forms/dmca, error-free basis, third-party providers, developer agreement, non-public areas, specific policies, /cards/overview, royalty-free license, /overview/terms/agreement, /web/overview, /en/periscope/super/terms, non-exclusive license, developer policy, /articles/15358-how-to-deactivate-your-account

RAKE's rankings between the scores of 4 and 3 inclusive: illegal conduct, twitter granting, /web/sign-, twitter cards, rights granted, commerce services, phone number, material subject, services twitter, account information, covered services, copyright@twitter, legal entity, unauthorized access, account, privacy policy, materials uploaded, collectively referred, venue clauses, content posted, content obtained, deactivate, displaying content, twitter trademarks, terms provided, receiving services, federal law

RAKE's rankings between the scores of 3 and 2 inclusive: periscope, twitter services, twitter terms

[3] Some of these terms are links and articles leftover from parsing Twitter's ToS

RAKE's rankings between the scores of 2 and 1 inclusive: access, content, twitter, entity, terms, sign, law, materials, account, services, material, clauses, services, conduct, rights, terms, /privacy, number, privacy, collectively

4.3 System 2

4.3.1 Nearest Neighbor vs Ground Truths

Company	F-score	Precision	Recall
Amazon Alexa	73.95%	68.38%	80.50%
Atlassian	54.51%	38.54%	93.09%
Instagram	98.99%	100%	97.99%
Reddit	79.62%	87.72%	72.89%
Rovio	72.91%	86.19%	63.17%
Twitter	69.77%	53.57%	100%

Table 3: System 2 ROUGE-1 scores Against Ground Truths

The metrics indicated in table 3 were the result of training with all documents except the one testing was being conducted with. Training was handled in this way to mimic leave one out testing. In leave one testing, we usually only leave out a data point, which is reserved for later testing. We shall call what we're doing, leaving a document for testing, leave one out testing also.

In this scenario, using leave one out for our training and testing pipeline was further necessitated by need for the context of the document while ensuring a clear separation of our training and testing data. Utilizing the leave one out methodology ensures this. As a result, the test data file was not be used until the supervised model(s) finished training.

4.3.2 ROUGE Metrics – Nearest Neighbor Summary vs. Member Rovio Ground Truths

Member	F-score	Precision	Recall
Shoshana	68.53%	86.20%	56.88%
Chris	46.89%	85.35%	32.33%
Andrew	20.93%	81.56%	12.00%

Table 4: System 2 ROUGE-1 Against Member Rovio Ground Truths

Table 4 shows model 2 training with every file except Rovio to create an output file for Rovio. We can't use Rovio for the training files since then we won't have a untouched test file. We compared that output file to each of the summaries we generated.

4.3.3 1-Nearest Neighbor vs SMMRY.com

Company	R	P	F-score
Apple	100%	12.96 %	22.95 %
BBC	100 %	3.18 %	0.061 %
Bing	98.30%	10.97 %	6.170 %
Blogspot	98.35%	11.32%	20.30 %
CNN	100%	7.14 %	13.33%
CouchSurfing	100%	5.40 %	10.24 %
del.icio.us	100%	6.12 %	11.54%
eBay	100%	2.68 %	5.220 %
Flickr	100%	10.89 %	19.64 %
GoDaddy	100%	15.10 %	26.24 %
GoogleAnalytics	100%	5.32 %	10.11%
IFTTT	100%	7.04 %	13.15 %
IMDB	100%	9.96%	18.11 %
Imgur	100%	12.52%	22.25 %
LastPass	100%	7.12 %	13.30 %
Reddit	100%	6.72 %	0.1260 %
ResearchGate	100%	0.30 %	0.5920 %
Signal	100%	8.82 %	0.1622 %
Twitter	99.16%	3.90%	0.0751 %

Table 5: System 2 (1-NN) ROUGE-1 Recall, Precision, and F-scores using the SMMRY.com-ToS;dr Corpus. All documents above were ToS, and scores are given with a 95% confidence interval.

The metrics indicated in table 5 were the result of training using Chris's documents as ground truth, then classifying each of the documents in the SMMRY-ToS;dr corpus.

5 Analysis

5.1 System 1

5.1.1 Ground Truth Analysis

After comparing the System 1 summaries to the ground truth summaries, we found there was low agreement on which sentences are important. While both summaries were extractive from each respective ToS, the ground truth was created based on one person's preferences summarizing a ToS. Some human preferences included keeping first sentences, ignoring verbiage about 3rd parties, and sentences that describe user rights and restrictions. System 1 had no such features to emphasize those sentences. While these approaches to summarizing are quite different, the low variation of the

scores shows consistency between how the ground truth was created and how the System 1 summary was created. Also since the F-Scores average to 34.47%, there's a semblance that both recognize similar core elements to be good summarizing sentences.

5.1.2 SMMRY Analysis

Comparing System 1 to SMMRY.com, the summaries have very low variation besides 2 outliers. Within Apple's Membership Summary, that score received the lowest score. Upon visual inspection of the differences of System 1's output versus SMMRY.com's summary, System 1's output kept uncharacteristically short sentences. This may be attributed to a broken parsing of sentences rooted in the extraneous newlines breaking certain sentences. For the very same reason, Academia's ToS also receives a low score. From these sentence fragments, a low recall comes from the unaligned summary sizes. Because the ranking algorithm was skewed from the sentence fragements, there was also a low precision in picking important summarizing sentences.

5.1.3 TOS;DR Analysis

When comparing System 1 to a manually created summary based off the legal analyses from TOS;DR, the ROUGE is comparable to SMMRY.com. However, there were some key differences comparing the focuses of each summary. One aspect System 1 diverges from TOS;DR is that System 1 will value longer sentences because they contain more information which gives it a higher rank from the core LexRank algorithm. TOS;DR aims to be more legal specific. For example, while TOS;DR highlighted age restrictions in their analysis summary, this sentence was not ranked high because System 1 has no awareness of the legal importance of age restrictions.

However, System 1 still scopes the summary well due to the general nature of the language of a ToS. ToS sentences that cannot be compressed without losing meaning tend to contain very specific wording protecting a company. For example, the 73 word sentence describes Twitter's ability to terminate accounts through all circumstances and for any reason at all. Since this sentiment was thorough in its wording, System 1 picked it up as important. This is arguably a viable summarizing sentence because a user would not have known about the large risk of losing their creative property to Twitter otherwise. The rest of System 1's summary is both verbose and is spanning language designed to be legally full proof.

5.1.4 RAKE Analysis

While there was no metric measuring the value of phrase highlighting, the results were analyzed. It was found that the RAKE valued unusual noun phrases and website links at the top. While useful in other applications, these feature did not scope well to the problem of summarization of a ToS because links are not informative alone.

However, the rest of RAKE's ranked terms fortunately contains important legal phrases. For example, RAKE found that "illegal conduct" was a key term describing the following Twitter sentence:

"TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE TWITTER ENTITIES SHALL NOT BE LIABLE FOR ANY INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR PUNITIVE DAMAGES, OR ANY LOSS OF PROFITS OR REVENUES, WHETHER INCURRED DIRECTLY OR INDIRECTLY, OR ANY LOSS OF DATA, USE, GOODWILL, OR OTHER INTANGIBLE LOSSES, RESULTING FROM (i) YOUR ACCESS TO OR USE OF OR INABILITY TO ACCESS OR USE THE SERVICES; (ii) ANY CONDUCT OR CONTENT OF ANY THIRD PARTY ON THE SERVICES, INCLUDING WITHOUT LIMITATION, ANY DEFAMATORY, OFFENSIVE OR ILLEGAL CONDUCT OF OTHER USERS OR THIRD PARTIES; (iii) ANY CONTENT OBTAINED FROM THE SERVICES; OR (iv) UNAUTHORIZED ACCESS, USE OR ALTERATION OF YOUR TRANSMISSIONS OR CONTENT."

In the original context, by highlighting that term in context, there is more emphasis on the important parts softly summarizing the meaning of the sentence.

5.2 System 2

5.2.1 Supervised Model Analysis

Model	Avg F	Avg P	Avg R
Perceptron	27.7445%	74.1248%	40.36%
SVM	34.9908%	86.098%	25.6486%
1-NN	74.9558%	72.3998%	84.6035%

Table 6: Average ROUGE-1 scores across perceptron, SVM, and nearest neighbor generated summaries compared to the ground truths.

Table 6 shows the average ROUGE-1 f-score, precision, and recall metrics for the summaries generated with leave one out training and testing compared to the ground truth. We compared that output file to each the summaries we generated.

System 2 implemented three supervised models at the beginning: k-NN, perceptron, and SVM. The ROUGE-1 metrics found in Table 6 were compiled after running each of the three supervised ML models on the ground truths, using leave one out methodology. The ROUGE-1 F-score, precision, and recall scores were then averaged out for each model across each Terms of Service document within the ground truth corpus.

In an initial comparison, it appeared that the resulting average ROUGE-1 precision scores were quite high across all three models. However, upon further analysis, it was realized that one particular model yielded scores that held an edge over the corresponding ones of its peers. Whereas both of the linear models (that of the perceptron and the SVM) failed to produce an adequate F-score, the k-NN model outperformed both of the linear ones implemented.

As it was evident that that out of three supervised machine learning algorithms implemented, we proceeded to seek out as much of a competitive edge in the supervised model as we could by tuning the number of neighbors used in the nearest neighbors algorithm. After tuning the k, we found that $k = 1$ resulted in the best performance. As a result, our primary machine learning model used for further experimentation was the nearest neighbor model.

5.2.2 Analysis of Results

We used two separate sets of ground truths in our model analysis.

The first set of ground truth consists of the hand curated summaries several team members individually created at the beginning of the semester based off of the Rovio terms of summary.

The second set of ground truth summaries used in the testing of the supervised models were manually created by team member Chris Tang.

Both sets of ground truths are extractive summaries from the original texts. That is, a sentence was decided whether it belonged or did not belong within the summary, with the resulting summaries being composed of all the sentences that had been decided belonged. Furthermore, both of these ground truths were sparse – that is, there was a lack of hand annotated data. Despite this shortcoming, accuracy prevailed when conducting tests by again utilizing leave-one-out by leaving the Rovio file out of our training data.

We primarily used the ROUGE metric to measure the performance of our models. In the analysis of the breakdown of ROUGE-1 F-score metrics, we pick apart the F-score into the individual ROUGE-1 recall and precision scores. For the sake of our analysis and final results, we place more weight on the recall metric than that of precision. Our reasoning is that the recall metric is an indicator of how much ground truth is in the model generated summary. It follows that we would rather the user read extra sentences, so long as the model generated summary contains all of the ground truth, than have the end user read a concise summary containing only a partial ground truth. Of course, that is not to say we neglected our precision metrics – we still ensured that the resulting numbers were satisfactory.

The gamut of metrics obtained from the leave one out analyses(Table 3) conducted was surprisingly ideal, as the datasets used in training were sparsely populated relative to other corpora used in training models. Despite this shortcoming, a high majority of resulting metric percentages were above 50% by varying levels. The resulting scores indicated that the supervised model recognized what Chris valued within a ToS, which is what we based most of training data off of. The resulting high scores indicate that we had some success in emulating this pattern.

The results(Table 4) when testing the nearest neighbor model on the Rovio summaries various team members generated were less than ideal. Only one team member's ROUGE-1 F-score was higher than 50%. A possible cause of this is due to the set of features implemented and used for the

supervised models. A majority of the features we choose were based on document length and paragraph length. Andrew's summary was the shortest out of all the group members. Therefore, the model would generate a summary that wouldn't match well with Andrew's. Chris's summary's length was between Shoshana's summary and Andrew's summary. The model matches a bit better with her's than Andrew's but not as good as Shoshana's. However, since our precision is high for all of these, but our recall is low as well the summary the model generated doesn't get all the sentences in the ground truth but most of the sentences the model did generate are in the ground truth.

The results in Table 5 when testing the nearest neighbor model on the SMMRY-ToS;dr corpus indicate almost perfect recall scores, but remarkably low precision scores. As a result, the F-scores for this dataset were also remarkably low, given how they are calculated and weighted. This actually happened because the summaries which were generated by System 2 were significantly longer than the flat seven-sentence summaries generated by SMMRY.com, but still contained all or most of the sentences generated using SMMRY.com. Overall, the results from testing against the SMMRY-ToS;dr corpus reveals that, given a ground truth based on the features which we want to emphasize, the model successfully classifies the desired sentences correctly (as indicated by the recall scores). The desired sentences however, are not necessarily reflected in the output from SMMRY (as indicated by the precision).

6 Team Organization and Contributions

Our team was organized into two sub-teams, each implementing a separate model. Each team member's individual contributions to their model, as well as contributions to the rest of the project are listed below.

- Mitchell Falkow
  - Scripted and performed collection of data through web scraping company ToS web-pages.
  - Cleaned retrieved ToS by hand
  - Scripted sentence segmentation of ToS
  - Generated the *SMMRY-ToS;dr* Corpus
  - Implemented various evaluation metrics including BLEU for usage (this part was abandoned in favor of ROUGE)
  - Implemented SVM model
  - Error Analysis of Model 2
- Ji hann Hong
  - Rovio ToS ground truth (hand annotated)
  - Implemented the System 1 Data Structure
  - Implemented RAKE and Keyword storage
  - Created Keyword Highlighting feature
  - Extracted the TOS;DR summary (manual extraction)
  - Ran the SMMRY vs System 1 Rouge Score testing
  - Implemented ROUGE scoring
- Andrew Ma
  - Rovio ToS ground truth (hand annotated)
  - Created python script to convert txt to special txt we can read features from
  - Created the features used to train the model
  - Created python script to read the special text and makes a json file with the sentences, features, and ground truth if it exists
  - Created models K-NN and Linear perception
  - Turned K-NN to 1-NN
  - Error Analysis of Model 2
- Shoshana Malfatto
  - Rovio ToS ground truth (hand annotated)
  - Sentence compression & preprocessing
  - General LexRank implementation
  - Ran the Ground Truth vs System 1 testing
- Chris Tang
  - Rovio ToS ground truth (hand annotated)
  - Assisted in feature creation
  - Created additional ground truths for both models manually
  - Implemented ROUGE scoring on perceptron, SVM, and k-NN generated summaries
  - Error Analysis of Model 2

7 Future Work

7.1 Improving System 1

While System 1 gives a decent summary, each sentence is still physically daunting to parse and understand. Due to this problem, a future improvement that could improve the output of System 1 would be a secondary compression layer that finds the root phrases and clauses. Then, in order to preserve meaning, these core phrases are emphasized through highlighting the way RAKE words are highlighted in a sentences. This would improve readability while retaining the meaning of each sentence.

7.2 Improving RAKE

RAKE can be improved by re-scoping the algorithm to emphasize key clauses instead of key phrases. With this key clause emphasize, RAKE could be applied to highlight intra-sentence phrases in order to further comprehend extracted ToS sentences.

7.3 Improving System 2

These initial metrics prove promising, however, there is much that can be improved. Due to a massive significant shortage in both training and testing data, the two ground truths that were used to test and train is lacking. Relative to the vast corpora usually incorporated when conducting such models, the credibility of the resulting ROUGE metrics is questioned. Thus, a vast improvement to the results of System 2 would be re-routing corpora through the pipeline detailed in this paper, the results of which would be less contested.

With regards to our failure in successfully training the models on the SMMRY.com-ToS;dr corpus, we note that further testing with a human-annotated corpus would provide better results, and would reveal more about the performance of the model.

Additionally, the features extracted from the data and used in the training/testing of the data were not of the best quality. The features implemented were rudimentary and were largely based on the location of the sentence relative to the overall text. We failed to incorporate any features that relied on word embeddings or the actual contextual meaning of the sentences. Any work attempting to further the results and models outlined in this paper would greatly benefit from using a different set of features, of which a subset should consist of drawing from word embeddings.

8 Conclusions

It is evident from the results outlined in this paper that both Systems 1 and 2 possess plenty of room for improvement, given the lack of data and dearth of quality features. However, we prospect that some headway had been made, as outlined here. The ROUGE-1 F-scores hovered around the $30\% - 44\%$ range for the results of Model 1. Taking into account the larger picture that is text summarization, these scores serve as a decent baseline for future work to be conducted on. The ROUGE-1 F-scores exhibited by the supervised 1-NN model hovered around the $60\% - 90\%$ range. Although these scores show great promise, it is important to realize that the features and data were can be improved with more training data, and research.

Acknowledgments

We would like to thank Dr. Heng Ji for her valuable contributions with regards to feedback, and guidance in the realization of this paper.

References

SMMRY.com. https://smmry.com/. Accessed: 2019-04-10.

Terms of Service; didn't read. https://tosdr.org/. Accessed: 2019-03-21.

tl;drLegal. https://tldrlegal.com/. Accessed: 2019-03-17.

Twitter. https://twitter.com/en/tos. Accessed: 2019-04-26.

Michal Belica. sumy (python library). https://github.com/miso-belica/sumy. Accessed: 2019-03-17.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA. ACM.

Google Research Datasets. 2017. sentence-compression. https://github.com/google-research-datasets/sentence-compression.

Bonnie Dorr, David Zajic, and Richard Schwartz. 2003. Hedge Trimmer: A Parse-and-Trim Approach to

Headline Generation. In *Proceedings of the HLT-NAACL 03 Text Summarization Workshop*, pages 1–8.

Günes Erkan and Dragomir R. Radev. 2011. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *CoRR*, abs/1109.2128.

Jade Goldstein, Mark Kantrowitz, Vibhu Mittal, and Jaime Carbonell. 1999. Summarizing Text Documents. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR 99*.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Leonard Richardson. BeautifulSoup (Python library). Accessed: 2019-04-10.

F. Rosenblatt. 1958. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386408.

Wei Xu and Ralph Grishman. 2009. A Parse-and-Trim Approach with Information Significance for Chinese Sentence Compression. In *Proceedings of the 2009 Workshop on Language Generation and Summarisation (UCNLG+Sum 2009)*, pages 48–55, Suntec, Singapore. Association for Computational Linguistics.