**What does the system do?**

This project tackles text summarization. The system summarizes tech news articles from the BBC Datasets.

**How did I tackle the task?**

The task was tackled in several steps. The first stage aimed to gain an understanding of how summarization is done. For this, I consulted different sources on how to build summaries. I wanted to know what makes a good summary. In short, a good summary conveys the same idea as the original text but using less words. I initially thought to have the summarizer summarize small texts that I would personally annotate. I read some texts and then manually tried to make a summary of them, to get an idea of the process. After I wanted to translate this knowledge and process into the computer system. To make the system understand what the text is about and to extract the most important parts of it. This task of making the system understand was more difficult than what I expected (see other section) so I tried a different approach. I decided to use the BBC Datasets given that they were readily available and had different news texts. I found that news articles contain the most important information at the beginning [cite]. The information is in the beginning because readers might not finish reading an article they started. With Zipf’s law I thought reasonable to produce the summary from a subset of the data of the news article. The subset of the data would be comprised of the headline (title), leading sentence and first 20% of the article (percentage was chosen because it appears in Pareto’s principle). The method for choosing the subset of the data was not completely researched, but it felt right. Additionally, the chosen subset of the data would hopefully contain the most important information about the article. As well, working with a subset of the data would reduce the complexity of the problem by projecting the text into a smaller dimensional space.

The subset of the data would then be passed through a pipeline that would give more information about the text and its structure. The initial pipeline consisted of several steps. First, separating the text into different sentences. Then tokenizing the text. After, perform POS tagging. Then, perform constituency parsing and dependency parsing. With the information of the pipeline, the summary would be produced. Here, trying to identify the most important parts of a text from the resulting representations in each step of the pipeline proved to be difficult (see section). I opted to discover some heuristic that could produce a summary from a text. For the heuristic, some elements of the pipeline were modified. In short, the heuristic worked upon the result of the constituency parse. The results of the constituency parse were chosen because I thought it gave information about the structure of the sentence on a sentence level. The system would not alter the information from the text, but extract parts from the original text to present as a summary. The design of the heuristic was based on manual inspection of the tech news articles of the data set. I noticed that news articles contain additional details that are not necessary for understanding the gist of the article. The devised heuristic aims to reduce these additional details from the text. To understand the parts that were not going to be included in the summary I consulted Stanford’s [cite] parser with some example sentences. What I was able to gather from the manual inspection is reflected in the heuristic. The devised heuristic is as follows:

[Include heuristic in pseudocode]

would be to get the first noun phrase of the sentence. By the first noun phrase it is referred to get the noun phrase that is closest to the bottom of the tree. This was beacuse the main parts of a sentence are the subject and the predicate. So in the summarizer the idea was to have that a summary of the most important parts of a sentence. It was done at a sentence level. So, in each sentence it was desired to get a noun phrase and an accompanying verb phrase. For the verb phrase the heuristic that the program followed was the following. Place everything that is under the tree until the first noun phrase is found. After the first noun phrase is found include

**System’s performance**

**Evaluation results**

**Manual inspection of output**

**Problems**

Several challenges emerged. How does the, first in my lack of linguistic terms I found it more challenging to do certain things, to understand what is the relation between certain objects in a sentence. I could intuitively summarize the texts, but it was difficult to replicate the process that one does to summarize and replicate in t in the machine. Because one has a some previous knowledge about what is being toalked in the article and can infer the missing details. The initial idea was to copy the standard way that I would approach to do a summary and then replicate it in the machine. To understand. The how to read a book was consulted. Of course, as mentioned it proved to be, at least at superficially, more difficult to make understand the machine language, than what the time permitted to do, and my current knowlede allowed.

Of ourse, if the constituency parsing was not correct, then the meaning could change and could not necessarily lead to to the best result.

**Future work**

Form the news articles there were different domains and the idea was to have, a type of classifier that to have some sort of bagging or boosting( don't remember) or summarier of texts. Some sort of biased summarizer for the different domains and then have each of the summarizers vote on which prhrases are deemed as important..

The following sentence does not work very well with the system: Tata Teleservices is using the lasers to make the link between customers' offices and its own core network. The laser bridges work across distances up to 4km and can be set up much faster than cable connections. In 12 months the lasers have helped the firm set up networks in more than 700 locations.

In some examples it the system and the approach seems to work. Nevertheless, I do not know precisely if it can be quantified. In the course we saw BLEU as a metric, but that it is a similarity metric. It could be that there is a group of gold-standard a annotated summaries and then the outputs from the system are given to the system. Additionally, . But this would only measure the similarity of the result summaries and the gold-annotated ones and not actually if they convey the meaning that is being wanted. In this case, I think that human evaluation could be better suited for evaluating how good the system is. Nevertheless, I can extract some examples from (the code does not work completely well, provided) of whwat the output could be. The evaluation results are deemed by me. Since there is no available metric that I believe that accurately would represent the behavior of the system. So at first one clear problem with the herustic is if the sentence or the construction of the sentence does not follow the pattern from which the heuristic was taken from. The heuristic might not extaract the most important parts of the sentence if the sentence or the text if it isn't in the correct structure. For example, in the sentence: the algorithm return something of the sort. And it does not actually convey very well the information. Another thing, is that the model has the bias that and the assumption that the most important information is present in the 20% of the article soit might not. If the assumption didn't get the output. Anotherissue is the fact that be that the sentences, since the system is not lreally understanding the way that the sentence is constructued, no mechanism of knowing what is happening. Then the sentences that are extracted by the system might be actually completely disconected from ech other. Another possible issue that I see wih the system is that the output of the system relies heavily on the input of it. This means that a poorly-written text leads to an output that is also poorly written given that the summarizer only takes fragments of the constructions persent in the sentence.

For improvements and extensions there are several alternatives.

Talk more about the challenges of building a successful LP system.

Extensions there are several. First, I believe that a bit more of linguistic knowledge could get one further ahead into this rather easy task, on the surface, of creating a summary. Additionally, I was thinking that the summary also depends on the target audience what the target audience wants, what some people might consider as a relevant in some aspects might not be relevant to other people. Then, if somehove the system could tailor the summaries based on a prototype of the person that is going to read the summary. That the system is aware of the person that is reading the summary. . Then, another improvement is the one mentioned previously of having several weak" summarizers that are able to summarize well a text from a given domain and then having a consensus algorithm that could identify the segments that could be the most important. Another insteresting aspect that I would like to explore is the concept of dependency parsing. In dependcy parsing the pependencies of the different words are presented and with this I believe a deeper understanding of the structure of the sentence and of the text could be understaood. The idea would be to have a meachinsm that could identify the object twhich other objects depend most on and then, hopefully this object is the most important object of the text. Having some sort of graph representation of all the text and to extract only the nodes that have the most connections and from those nodes with the most connections build a summary. The problem of summarizing a text, although simple at first sight, at least in human terms, read the text and thhen condense the text leaving only the most important parts, like cooking when you do a reduction and the ingredients get embedded into the other ingredients. The task proves to be more difficult when actually trying to implement it. The challenges of how to quantify understanding I believe is difficult, to have a way to formalize all the knowledge that we take for granted as language I thought was rather astounding. In a way it made me realize how complex language and the production of language can be, and such a thing that we take for granted is very complex and yet we do not grasp the complexity of it, at least in my case. The task served to face the challenges that are present in the field to realize that it is not so simple as it sounds. That dealing with meaning is difficult. A little bit more of perspective of what NLP is. Additionally, if I had more time I would like to do a better survey of what the existing systems and state of the art solutions exist for doing text summarization. I used partly what was learned from the course to guide myelf into. Another realization is that you canhave the pipeline and then you can process pass the text into everything in the pipeline but that does not necessarily give out the parts you need to build your solution. In a way a way of seeing that tools in this domain, though useful, are not magical, and that they they e a tool for the person that is using or developing the system. A tool for

Another problem would be in a conversation, with all the baggage that comes with analyzing conversations, the stops and that at times the sentences might not be constructed in a certain manner, that they do not follow the standard structure.

**References**

D. Greene and P. Cunningham. "Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering", Proc. ICML 2006.