Large Scale Text Analysis

Kevin Tee
Xinchen Ye
Weijia Jin

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2015-129
http://www.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-129.html

May 15, 2015

Acknowledgement

We would like to thank Prof. Laurent El Ghaoui for advising us during the span of our Capstone project. We would also like to thank StatNews research group we are a part of for giving constructive advice and technical support.

University of California, Berkeley College of Engineering

MASTER OF ENGINEERING - SPRING 2015

Electrical Engineering and Computer Science

Data Science and Systems

Large Scale Text Analysis

WEIJIA JIN

This **Masters Project Paper** fulfills the Master of Engineering degree requirement.

Approved by:

1. Capstone Project Advisor:

Signature: ___________________________ Date _____________

Print Name/Department: LAURENT EL GHAOUI/EECS

2. Faculty Committee Member #2:

Signature: ___________________________ Date _____________

Print Name/Department: LEE FLEMING/IEOR

Abstract

We take an algorithmic and computational approach to the problem of providing patent recommendations, developing a web interface that allows users to upload their draft patent and returns a list of ranked relevant patents in real time. We develop scalable, distributed algorithms based on optimization techniques and sparse machine learning, with a focus on both accuracy and speed.

Table of Contents

*Co-written with Kevin Tee and Johanna Ye

Problem Statement

Small businesses often face many challenges when filing for a patent and searching for relevant patents, from the threat of potential litigation from large corporations to costly lawyer fees. Currently, most patent-related tasks are often done by lawyers, an option that is both expensive and time-consuming. Lawyer fees can often cost up to thousands of dollars, although their services take on a broader range: from understanding the patent filing process to domain-specific knowledge to patent recommendations. We aim to automate the process and expedite the search process by helping researchers and inventors determine if their technology development or unique idea has been previously patented. We take an algorithmic and computational approach to this problem, developing a web interface that allows users to upload their draft patent and returns a list of ranked relevant patents in real time. With the growing amount of data available, as well as computing power and fast storage and retrieval, taking a data-driven approach can utilize this data to solve difficult problems, especially those related to text. Text mining technology can be directly applied to completely automate patent matching within the patent database. This technology has a wide range of applications in many different industries, from aiding fellow researchers in finding the information they need, to providing data analytic tools for working professionals from any field or domain (Smith, 2014). Because we lack substantial computing power, we are forced to think about algorithms that can scale without the the use of large machines or GPUs. We develop scalable, distributed algorithms based on optimization techniques and sparse machine learning, with a focus on both accuracy and speed. Our methods rely on both state-of-the-art research in academia mixed with our own novel approaches. Currently, we develop an unsupervised approach to this problem, although recent we have investigated semi-supervised approaches with much promise. Our method is targeted at small businesses who are searching for a cheaper alternative to lawyer services. Not only do we offer a low-cost alternative for small businesses, but our algorithms are real-time, and return results in a few seconds. As more users use and provide feedback based on the relevance and ranking of our results, the algorithm will improve (again, in this semi-supervised setting), allowing us to iterate on our current prototype integrating a user feedback loop.

Strategy

This section will address our strategic positioning from a market perspective, touching on our relationship with customers and suppliers, our “go-to-market” strategy, and industry trends. Additionally, it connects these ideas to those of our competitive landscape, discussing industry potential and our competitive advantage. The first part of this section will investigate the market and our competitors as well as the barriers to entry. The second portion of this section will do a more in-depth analysis of the industry, examining the patent industry and our differentiating factors from the lense of several of Porter’s five forces. At the end, we will look at the various trends in both the patent industry, and how it affects our short and long term strategy. Those components of the industry analysis will present a unifying view of our market strategy and the factors which accompany it, integrating the Five Forces model throughout the paper (Porter, 2008). We start by analyzing the competitors and potential substitutes for our product.

Two substitutes, Google Patent Search and the USPTO website, do not pose a large threat to our product because they lack a quality algorithm and a robust, informative UI giving our product a unique competitive advantage. Since the market and industry are intertwined and dependent on each other, a brief introduction of the market can shed light on the industry we are in and the substitutes or solutions in this space. Our primary target market is small businesses that have very limited resources but want to protect their technology or avoid using patented technology by providing effective patent similarity searches (we discuss this more in subsequent paragraphs). Our industry is within the software industry that helps businesses protect their IP and avoid litigation incurred by using patented technology. In this context, three major substitutes/alternative solutions deserve our attention and analysis: United States Patent and

Trademark Office (USPTO), law firms, and online patent searching services (e.g. Google). For each of these substitutes, we describe what about our product offers that the other does not, or our competitive advantage.

The USPTO presents an obvious substitute for any patent search product, providing a free web search interface that is alluring to small businesses due to the cost and the brand name; however, the USPTO's search functionality is limited in a few key ways. The USPTO reviews trademark applications for federal registration and determines whether an applicant meets the requirements for federal registration. They have the most up-to-date and comprehensive database regarding all of the patent data, but the website does not answer questions regarding whether a particular invention is patentable and that is also difficult to use and navigate. They only offer keyword search (e.g. "Machine Learning") but not document search with similarity ranking, and often when small businesses are looking to do a patent search, they want to compare it against their draft patent. For the purposes of determining patent similarity, the USPTO serves more as a supplier than competitor. Our technology is meeting an unmet need that the USPTO doesn't satisfy, and in the process making it more convenient for small businesses. We provide much better functionality (as well as a more user-friendly web interface), rendering USPTO a weak substitute/competitor in the dimension that our product will compete.

The other substitute/competitor is the law firm which facilitates the patent filing process, but our product is significantly cheaper while providing a similar, if not better, quality of service. The law firm can carry out multiple functions: it can help determine if an invention qualifies for a patent or potentially infringes on an existing patent; it can guide the customer through the patent application process, and the law firm can work with the customer to enforce the patent against potential infringers. However, our research shows that a professional patent search may cost approximately $500 to $1,500, while obtaining a patentability opinion from a lawyer costs approximately $1,000 to $2,000 (Costowl.com, 2014). Additionally, setting up an appointment with a lawyer is a slow process and often requires multiple visits. Our product addresses both of these issues by providing a cheaper alternative for small businesses and giving near instantaneous results. We are able to achieve these near instantaneous results through efficient indexing and parallel algorithms on the backend of our website. If there was an online service that provided a patent similarity match and gave instantaneous results, the response speed and the reduction on cost would provide immense efficiency and reduce the burden on budgets for small *and* large businesses alike. However, our service can be somewhat subpar to law firms as we don't provide the personalized "touch" or face-to-face interaction with a small business. There is an inherent tradeoff that a small business owner must make, and we've decided to segment the market such that we enter at a cost-effective end. Our product may not outperform every aspect of a law firm, but for our target market, small businesses, it is the most sensible option.

Our third competitor is the online websites for patent searches, and while these services are most close to the service we aim to provide, our product holds a unique competitive advantage. The most publicly-known search is Google patent search. According to Google, their patent search ranks results according to their relevance for a given search query without any human oversight. However, Google Patent doesn't support document search and the algorithm is not transparent. Our project differentiates itself from Google Patent in that our functionality supports document similarity ranking and provides strong interpretability that helps the user qualitatively understand the results. Our goal is to help users understand the results that are given back to them: make sense of which parts of each document are most similar, as well as ask the user to give some feedback on the quality of the results. This gives us to the ability to improve our algorithms by integrating user feedback and improving our results with time. In a sense, what we are building is not just a search engine, but a interactive website to help small businesses that is adaptive and gets better overtime.

The threat of new entrants is of a moderate concern, although it requires time to establish oneself in the market. Given that the particular industry doesn't have a strong force from the supplier and the ease of getting the data to start a new online service with patent ranking functionality, we note that the barrier to entry is low, as with most open source software projects. However, the quality of the algorithm and performance under large amounts of data is one of the most important differentiation factors from one online service to another. It takes time to aggregate data, discover the optimal solution for data storage and iterate on the algorithm with more user data. Additionally, to protect our leading position and our state-of-the-art technology as the incumbent in the space, we will consider filing patents for the database configuration and algorithms that we are using in an effort to raise the barrier of entry for new entrants.

The interpretability of software products impacts the user's decision for choosing the product. Early entry allows us time to fine tune the algorithm and develop customer relationships. With early customer relationships, we can develop demand-side benefits of scale. As more customers use our technology and advocate it, this in turn will generate more new customers. The more customers we have, the better testing data we can obtain. With the aggregation of data, we develop better products and the positive iterative circle continues. Over time, as more customers use our product and we apply interactive feedback, the greater success we will achieve. Developing this feedback loop is crucial to our success and helps to prevent the new entrants who have not been in the industry long enough to aggregate user data.

As noted above, the major challenge and potential competition in our industry is the quality of algorithms and customer relationships. Threats of competition, both current and future, is relatively low; however, to maintain our dominance requires us using our position as incumbents to constantly iterate and improve our algorithm, and to us our own Intellectual Property (IP) to raise the bar of entry for new competitors. Having established our competitive advantage in our industry and differentiating ourselves from various competitors, we turn now to analyze our customers and suppliers, and our relative positioning among them.

While the demand for patent recommendations is somewhat diverse, our target customers will be small businesses. The needs for patent recommendation and patent services span across many markets, ranging from commodities such as coffee machines to huge operations such as pharmaceuticals (MINTEL, 2014). Interestingly, MINTEL does not have a section of their reports on the patent market, but is a common topic in many articles, suggesting that patents are crucial in many markets. Currently, patent lawyers tend to drive the market, with both their expertise and ability to customize to their customers with services include patent application and renewal services, litigation services, and patent novelty search (Carter, ; Hoovers, 2014). Our product aims to compete directly with patent novelty search, with our differentiator being novel algorithms and speed, both of which we will discuss in the following section. The number of small businesses in the United States is approximately 23 million, accounting for 54% of all the sales, and they provide 55% of the total jobs in the United States since 1970. Most importantly, the number of small businesses has increased by 49% since 1982, suggesting its enormous growth and potential (SBA). Small businesses as customers exhibit a strong force in the context of Porter’s five forces because small businesses have many options for patent novelty search, in both lawyer firms, other search engines, and online resources. Porter argues that the industry structure drives competition and profitability, and in our market, the differentiating factor is the quality of service (in this case patent recommendations, reliability, algorithms) suppliers can provide (Porter, 2008). We therefore turn to weakening the forces of customers by looking at our competitive advantage in the context of a marketing strategy.

Designing a market strategy using the 4Ps model will give us the ability to take our idea and technology and offer it as a robust service (McCarthy). As stated earlier, our competitive advantage lies in our ability to provide apply novel algorithms to the problem of patent recommendations, with excellent results in both speed (time it takes to return a query) and accuracy (relevance). Along with law firms, there are other search engines for the same purpose: Google Patent Search and subject-oriented databases (Google, Carpenter). Currently, our product is a minimum viable product (MVP): we have a minimal, simple web interface with an algorithm that has huge potential to be improved with customer feedback. Our pricing model will be per-document based: for each draft patent, a customer will be charged a flat rate. This rate will be significantly cheaper than going to a lawyer, which can cost thousands of dollars (Carter, 2015). Promotion strategy will be largely dependent on our positioning within the market and the market segment of interest. While the market for small businesses is large (23 million total small businesses), not all of these companies are in need of IP or patents (SBA). For example, a local restaurant would not need our services. Instead, we define our market segment to be small businesses in the United States less than two years old with unfiled IPs or patents. The service will be promoted primarily through a single channel: advertisements on social media sites such as Facebook, Google, etc. Running a targeted ad campaign toward small businesses will help to spread the word about our service. Overtime, our goal is to establish a reputation and gain more sales and awareness from word of mouth (so future customers will be more drawn to our product). Distribution (place) is easy for us, as our service is a website (although we host our content on servers, more about that in suppliers) and is therefore easily accessible provided one has internet connection. With these 4Ps and our competitive advantage, we have positioned ourselves to weaken the force from the customers, allowing us to differentiate ourselves in a competitive market.

The two suppliers we will need are web servers and a patent database. The power, or force, of both of these suppliers is low, much to our advantage. The trend toward using web services (Amazon, cloud, etc.), along with cheaper storage and more competition, will drive down the price (Hart). Colloquially, web services is a service which allows website to run queries (such as a Google search, or patent algorithm). These web services tend to be run in large databases, or “data farms”, which are dedicated to processing large amounts of information or queries. Moreover, because our development is still in the initial state, switching costs are low and moving from one web service to another web service is relatively easy. Additionally, obtaining patent data from the United States Patent Trademark Office (USPTO) or Google Patent Search is relatively simple. Our program is able to “scrape” all the relative information from these websites and deposit the information into a database. Because these patents are available to the public, they will remain available and easy for us to use. The force from the suppliers, as with many web startups, is low and puts us in a good position going forward with our market strategy.

Despite the strong force from customers in Porter’s five forces models for our patent recommendation service, having carved out a particular subset of the enormous market helps to weaken this force. Along with the ‘go-to-market’ strategy using the 4Ps model, a weak supplier force, and analysis of industry and trends from the complementing papers, our service can immediately make a substantial impact in the market. In addition to understanding the forces from customers and suppliers and how to mitigate them, it is important to understand the current market trends related to our product. In particular, we look at technology trends related to machine learning and big data.

As research in artificial intelligence (AI) bears fruit, we are increasingly seeing its applications in everyday consumer products, from smartphones that can take instructions directly from voice commands to online retail websites that can make uncannily accurate predictions of what goods we need or what movies we may want to watch next. Patent search/recommendation is another one of those applications which took advantage of recent developments in natural language processing and machine learning techniques. What used to be a task monopolized by lawyers who made use of their expertise in patent law to help clients determine whether their new product or technology is patentable (USPTO, 2014) can now be accomplished using software at not only a much faster speed but also at a fraction of the cost (NOLO, 2014). This is possible all thanks to the technological trend towards smarter and more cost-efficient computing.

In the past, the sheer size of the patent database (numbering in the millions and increasing by the thousands each week) and the fact that patent documents are notoriously hard to decipher due to their abundance of legal jargon (USPTO, 2013), are a huge deterrence to the layman who wish to extract some information (needle) from all the patents (haystack). However this is no longer the case with the advent of intelligent search and recommendation systems that not only returns query matching results in the blink of an eye, but also directs the users straight to the relevant portion of the document, saving them the trouble of having to navigate through the dense text. The possibility of translating this previously exclusive domain knowledge of patent lawyers into computer algorithms opens up the opportunity for software to come in and contest the monopoly previously held by patent lawyers over the business of patent search (Carter, 2015). We plan to fully exploit this opening in the market brought about by new technology. By packaging our novel patent search algorithm as an online web service, we wish to market it as an attractive and widely accessible platform for any interested party to conduct their patent searches.

The business of patent search is very lucrative, particularly in the technology industry where new technical breakthroughs and their protection can make or break a company. Top technological giants such as IBM average 9.3 patent applications in a single day (IBM, 2014). This focus on creating and maintaining a technological advantage will only intensify with the proliferation of start-ups that are built on the latest technology. The huge opportunity presented here has attracted a lot of big players into this field including Google Patent Search, Espacenet and many others (Google Patent Search, 2011). As a new entrant, the competition posed by these established services and future entrants will be a big challenge. However since the advent of online patent search has been relatively recent, there is still room for us to mitigate the threat of intense competition by differentiating ourselves from our competitors by building smarter algorithms, a more intuitive user interface and more insightful data visualizations. By departing from the standard model of keyword search to document search, we will be able to serve an as yet unmet need of companies which need to be able to check in a timely fashion, if what they are currently developing infringes on any published patents. Another way we can reduce the effects of market saturation is to expand our customer base. The traditional customers of patent search services are mainly organizations who want to find a certain technology or to know whether their inventions can be patented, and can afford the costly lawyer fees (Lawyers.com, 2014). A new group of users who will be receptive to easily accessible patent search web services are researchers, students or the curious browser who wish to gather more information about a certain invention or technology, regarding its history, inventor(s), inspirations etc. We can reach them by marketing our product to university departments and through our research circles.

Advances in AI technology has made our product possible and has given us an opportunity to disrupt the industry of patent search. However, this has also created intense competition within the industry which we hope to overcome through product differentiation and customer base expansion. With our novel algorithm targeted at small businesses, we have found a portion of the market which we believe we can succeed, and slowly grow our customer base through word of mouth advertising. In addition to strategy, however, remains its counterpart: operational effectiveness. Our strategy and analysis using Porter’s model is not enough, but it does provide the necessary framework to be able to execute and ultimately carry out our vision.

Intellectual Property

In this section, we will describe our intellectual property (IP) and describe our choices that surround our IP. For a particular IP, there are several interesting options to explore, including filing a patent, making the work open source, etc. In our case, we have decided that while our algorithms are potentially patentable, we will not file a patent nor pursue any other alternate IP strategy. There are several reasons for this decision, which we discuss in the subsequent paragraphs. It is important for the reader to understand, however, that our decision not to pursue IP strategy is what we perceive to be the best choice in the present, and that it may change as our product and competitive landscape evolve.

The most patentable item of our project is the our algorithm for recommendations, but the modularity of our algorithm makes it inherently difficult to patent. An often cited example of a “good” and “patentable” algorithm is Google’s PageRank algorithm, filed by Larry Page during the birth of Google. In their algorithm, there is a very clear statement of how their patent differentiates current work, asserting that the number of links going to a particular webpage is proportional to its importance (Page). In our algorithm, however, while it uses state-of-the-art and current optimization techniques, is more modular by design. This means that our algorithm has many different components that work together (extracting topics from the corpus, running similarity metrics, etc.). Each of these components aren’t particularly new: they reference longstanding literature and papers in the field. We are able to achieve good results because of our creativity in putting these modules together, not because we developed a landmark breakthrough. One could argue that patenting the method of combining modular algorithms together would make sense. However, this would make our algorithm available for the public to build upon, which would put us at a competitive disadvantage. We discuss our reasons against this potential solution in the following paragraph. Additionally, investigation into the patent database shows that there are almost no patents, from a machine learning perspective, that do a similar "combining" of existing work in the software/algorithms field (Google Patent Search). This suggests that it will be difficult to patent our algorithm, likely because there is not enough of a differentiating factor.

A brief cost benefit analysis of patenting our algorithm shows that the resources invested will not provide significant returns, as well as open the door for competitors to see our "trade secret". Filing a patent is estimated to cost $2000, with $130 alone for the actual submission and approval of the patent by the United States Patent Trademark Office (USPTO). In theory, our primary motivation for filing a patent would be to protect ourselves in the case where an independent developer comes up with our algorithm. This however, is very unlikely because of the complexity of our algorithm and our use of ideas from different fields within computer science: optimization, probabilistic models, machine learning, and efficient database access and retrieval. Rarely do software corporations dispute over particular algorithms, but rather specific features in products such as touch screen display for phones (Bosker). This is because it is difficult to draw the line between the "same" and "different" algorithms, especially when there are many moving parts like our algorithm. A more pressing concern for us is the danger of leaking our algorithm to competitors. One nice aspect of developing a web application is that the algorithm is hidden, and there is no way for people to see how we provide the recommendations. Consumers (and competitors) can only see the final result, the ranking of the patents. If we file a patent, however, it gives competitors full access to our algorithm and be able to improve upon it. Again, because it is difficult to differentiate an algorithm, our patent may not be significantly similar to their improved methods and we would be unable to file a patent lawsuit. In our case, filing for a patent gives an unwanted consequence, in addition to the time and money incurred throughout the process. More succinctly, filing for a patent will reduce our competitive advantage in our industry and provide more of a hinderance than a benefit.

Keeping our IP secret will allow us to continue to iterate and improve our algorithms with more customer feedback. Part of our continued strategy moving forward is to use customer feedback and user studies to refine and tweak our algorithm. If we file for a patent, we are in a sense "binding" ourselves to that patent. In software engineering, a common practice is agile development, where small incremental improvements are made in 1-2 week "sprints". Having adopted this method for our group, we are able to make quick improvements because of the flexibility to tweak algorithms and rapidly iterate. As a result, filing for a patent will inhibit this structure and make it more difficult to innovate. Nonetheless, a big part of the software engineering culture is giving back by contributing to open source. Open source is the idea where software developers make their code readily available online, so that other developers can build upon this work and push the field of computer science (High). Some extend this idea further and assert that software engineers have an obligation to contribute to open source communities such as Github (Masson). While we will not be making our algorithm publicly available, we will make parts of our code available such as data parsing and database indexing, under a copyright, a free license for software such as BSD or GNU Public License. This is consistent with many startups and technology companies today, who do not place their "secret sauce" on a publicly available repository.

Our group's IP strategy is to not pursue a patent, copyright, trade secret, or other forms of legal recognition of our work. While we plan to open source some of our code, in align with the culture of software engineers, we plan to keep our algorithm secret. Our motivation for doing so fall into two large categories: the difficulty of getting our algorithm patented, and the consequences incurred as a result of this course of action. We view the latter as more damaging to our ability to produce a great, differentiating product because it releases our ideas to competitors and prevents us from innovating at a fast rate. However, as we move forward, it is important to keep our options open. If we find a new way to conduct user studies, for example, it might be worth filing a patent, as it is more concrete and less susceptible to misinterpretation. Similarly, if we decide to make more parts of our code open source, then we will accordingly copyright the work under some software license.

Technical Contributions

The goal of the project, Large Scale Text Analysis, is to build a novel search engine for US patents from the United States Patent and Trademark Office (USPTO). Google Patent Search already allows the user to query all patents by keywords, this is great if the user has a topic in mind and wants to do research on the current patented technologies in that area. We are approaching this from a different angle, what if the user already has a brilliant idea or a product he is working on or and he wants to know if there are any potentially infringing patents out there. By allowing users to upload their own draft patent or a description of their product or innovation, our search algorithm will return a list of patents similar in content to the query. Since the total number of granted patents is in the millions and that number is rising by the thousands each week, the main challenges of our project has been to optimize our algorithm for speed and scale and to evaluate the relevance of the output ranking.

Since building a search engine from scratch requires work on all aspects of the data pipeline, we have largely divided the work among us as follows: Johanna on designing the search algorithm for a small dataset, Kevin on optimizing that algorithm for larger inputs, and Weijia on data collection and the evaluation and visualization of search results. We have tried out many different metrics for determining the ranking of search results, and often various variants of a single similarity metric as well, hence it is important to come up with a way to compare the effectiveness of different methods against each other.

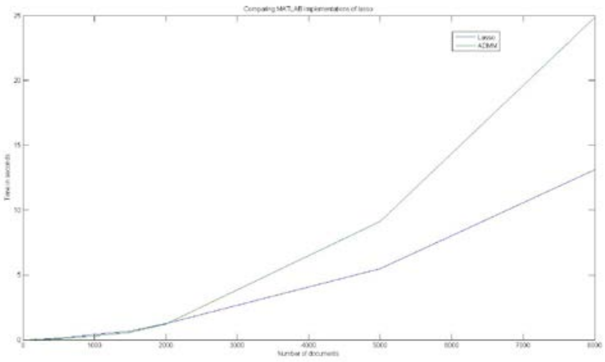
First, I will give a brief description of the source of our data and how it is stored in the database. Since the USPTO website only allows retrieving patent documents via search queries or specific patent IDs, it would be too slow and inefficient to obtain the entire database of patents by scraping the website directly. Fortunately we found an alternative download site maintained by Google which contains week by week worth of patents in compressed archive format. We set up a script to automatically download the archives in our selected time period, transform each patent document into a vector of words and eventually into a vector of numbers each representing the TF-IDF (term frequency-inverse document frequency) of that particular word in the document. This number is basically representative of how relevant that word will be when comparing document similarity across other patents (El Ghaoui, 2011). More details of this process and how the data stored in this format feeds into our subsequent lasso algorithm can be found in Johanna's paper.

As Kevin mentioned in his paper, one of the ways we speed up the execution of the algorithm is to do preprocessing on the stored data. While we use the lasso to find the most similar documents given the query in real time, clustering of all the words in the stored patents into their relevant topics can and should be done beforehand. Since this clustering step can take a long time especially for larger datasets, we precomputed the word clusters and the intermediate projective matrix (a matrix projecting data from the word space to the topic space) and stored them in an external python data structure. This way whenever there is a new query, we simply load the preprocessed data into the workspace and run the lasso step directly, the projective matrix is used in real time to transform the input vector into the topic space. More information on this clustering step can be found in Kevin's paper.

With preprocessing in place, the bottleneck of our algorithm becomes the lasso runtime, which provides the vital document ranking functionality of the search engine. Since lasso has to optimize the vector of weights (i.e. relevance) corresponding to each patent document in the database subject to regularization constraints, it performs much slower than conventional document similarity metrics such as cosine similarity and k-nearest neighbors. One method to resolve this that Kevin has briefly mentioned is to substitute our current lasso implementation with a parallel implementation that ideally, can improve the performance of the algorithm by distributing the computations required across multiple machines. A parallel method, called alternating direction method of multipliers (ADMM), is a distributed solver for convex optimization problems such as the lasso (Boyd, 2011). Its approach is to break the problem into smaller pieces, iteratively solve for separated variables using partial updates instead of iterating to convergence on a single objective function. The ADMM algorithm represents problems in the following form:

$$\text{minimize } f(x) + g(z)$$
$$\text{subject to } Ax + Bz = c$$

It replaces the constraint optimization problem with a series of unconstrained problems and a penalty term, each iteration of the algorithm with involve a x-minimization step, a z-minimization step and a dual variable update. Since the x-update takes the form of a ridge regression problem, ADMM is basically a method to solve lasso by iteratively doing ridge regression. The distributed version of the algorithm splits across machines by training examples, coupling the computations for data blocks by including collecting and averaging steps in each iteration.



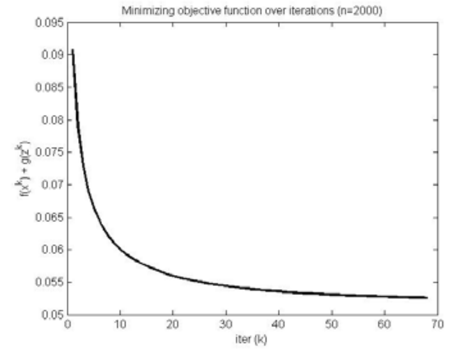


Figure 1 Comparing performance of lasso using standard MATLAB implementation vs ADMM algorithm (left). Convergence rate of ADMM algorithm on dataset size 2000 (right).

However there is a tradeoff between saving time by reducing the amount of work per thread and the addition overhead generated by the inter-process communication required to update all the threads with each other's progress. Figure 1 shows that even though the ADMM lasso converges at an extremely fast rate (just tens of iterations), its performance does not match up with the standard implementation of lasso for larger dataset sizes. This plot was generated on a single machine, hence we would expect much better speedups when running on a cluster of machines. Also since the full dataset is used in these experiments (pre word clustering), it is not yet clear what differences it would make when this algorithm is used in combination with topic modelling to restrict the length of the feature vector across all dataset sizes.

Since we have come up with various variations and optimizations of the search algorithm throughout the project, we need a way to help us judge which implementations perform better than others in ranking search results for the same query. This brings us to the visualization of search results, we built a user-friendly web interface for users to send their text queries and to compare the output from different algorithms side by side. This involves standardizing the output format across the algorithms, a search function to take in the query and the group of algorithms to run, building HTML templates and setting up a web server supporting RESTful APIs. Figure 1 shows the index page of our web application for search and figure 2 shows the display of top ranked results from two different algorithms for the same query. While this example currently shows only the search options for the lasso and cosine similarity algorithms, any other group of algorithms can be chosen for comparison with additional buttons. Right now the display page includes the patent title and abstract in order for us to compare at a glance, the top-ranked documents under each algorithm and using the information from the abstract, we can determine which output contains patents most similar in content to the query. We can also set the display page to only display patent titles in order to get a quick overview of which group or category of patents gets ranked the highest.

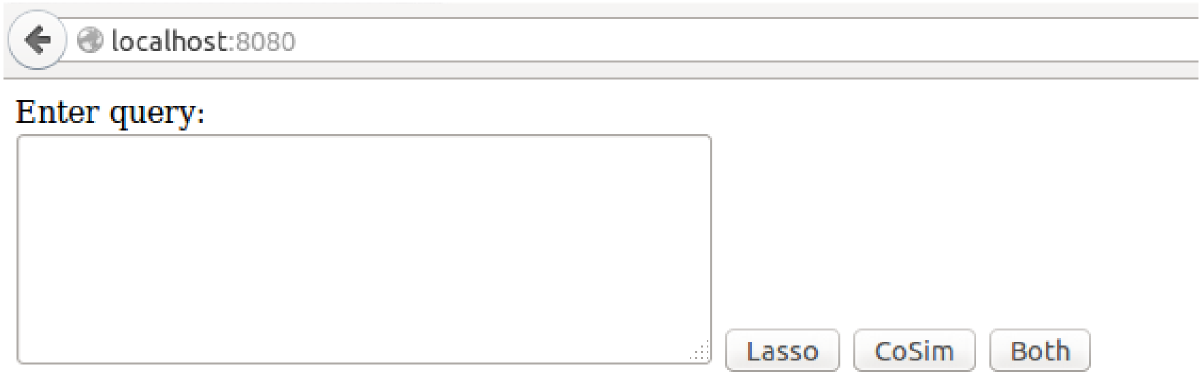
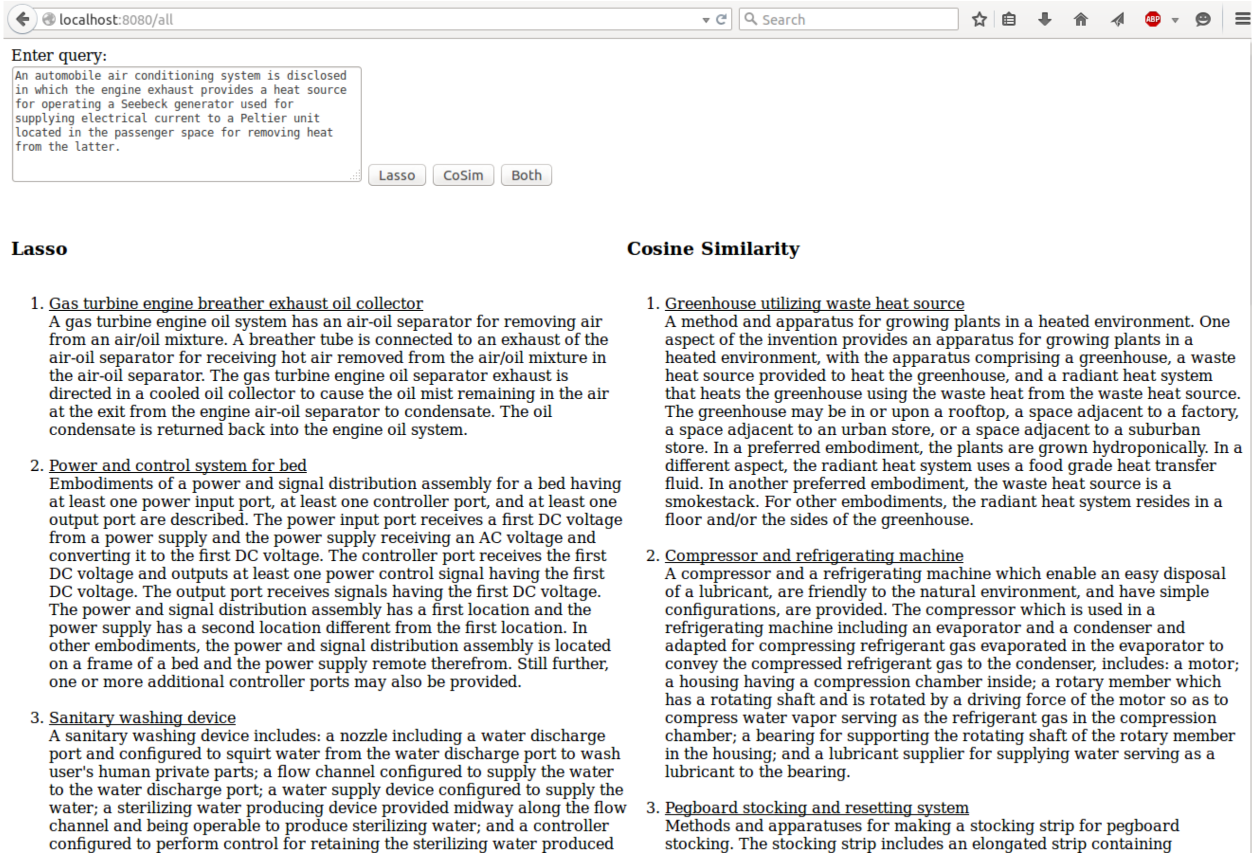


Figure 2 Index page of search application



Enter query:

An automobile air conditioning system is disclosed in which the engine exhaust provides a heat source for operating a Seebeck generator used for supplying electrical current to a Peltier unit located in the passenger space for removing heat from the latter.

Lasso

1. Gas turbine engine breather exhaust oil collector
   A gas turbine engine oil system has an air-oil separator for removing air from an air/oil mixture. A breather tube is connected to an exhaust of the air-oil separator for receiving hot air removed from the air/oil mixture in the air-oil separator. The gas turbine engine oil separator exhaust is directed in a cooled oil collector to cause the oil mist remaining in the air at the exit from the engine air-oil separator to condensate. The oil condensate is returned back into the engine oil system.

2. Power and control system for bed
   Embodiments of a power and signal distribution assembly for a bed having at least one power input port, at least one controller port, and at least one output port are described. The power input port receives a first DC voltage from a power supply and the power supply receiving an AC voltage and converting it to the first DC voltage. The controller port receives the first DC voltage and outputs at least one power control signal having the first DC voltage. The output port receives signals having the first DC voltage. The power and signal distribution assembly has a first location and the power supply has a second location different from the first location. In other embodiments, the power and signal distribution assembly is located on a frame of a bed and the power supply remote therefrom. Still further, one or more additional controller ports may also be provided.

3. Sanitary washing device
   A sanitary washing device includes: a nozzle including a water discharge port and configured to squirt water from the water discharge port to wash user's human private parts; a flow channel configured to supply the water to the water discharge port; a water supply device configured to supply the water; a sterilizing water producing device provided midway along the flow channel and being operable to produce sterilizing water; and a controller configured to perform control for retaining the sterilizing water produced

Cosine Similarity

1. Greenhouse utilizing waste heat source
   A method and apparatus for growing plants in a heated environment. One aspect of the invention provides an apparatus for growing plants in a heated environment, with the apparatus comprising a greenhouse, a waste heat source provided to heat the greenhouse, and a radiant heat system that heats the greenhouse using the waste heat from the waste heat source. The greenhouse may be in or upon a rooftop, a space adjacent to a factory, a space adjacent to an urban store, or a space adjacent to a suburban store. In a preferred embodiment, the plants are grown hydroponically. In a different aspect, the radiant heat system uses a food grade heat transfer fluid. In another preferred embodiment, the waste heat source is a smokestack. For other embodiments, the radiant heat system resides in a floor and/or the sides of the greenhouse.

2. Compressor and refrigerating machine
   A compressor and a refrigerating machine which enable an easy disposal of a lubricant, are friendly to the natural environment, and have simple configurations, are provided. The compressor which is used in a refrigerating machine including an evaporator and a condenser and adapted for compressing refrigerant gas evaporated in the evaporator to convey the compressed refrigerant gas to the condenser, includes: a motor; a housing having a compression chamber inside; a rotary member which has a rotating shaft and is rotated by a driving force of the motor so as to compress water vapor serving as the refrigerant gas in the compression chamber; a bearing for supporting the rotating shaft of the rotary member in the housing; and a lubricant supplier for supplying water serving as a lubricant to the bearing.

3. Pegboard stocking and resetting system
   Methods and apparatuses for making a stocking strip for pegboard stocking. The stocking strip includes an elongated strip containing

Figure 3 Results comparison for different search algorithms

In the example shown above in figure 3, we used a limited dataset of a month’s worth of data (about ten thousand patents) and the following test query: “An automobile air conditioning system is disclosed in which the engine exhaust provides a heat source for operating a Seebeck generator used for supplying electrical current to a Peltier unit located in the passenger space for removing heat from the latter.” Even with the limited time range of data, we find that the quality of the lasso ranking outperforms that of cosine similarity. The former provides patents on engines and electrical power systems as the top results whereas the top result from cosine similarity talks about a greenhouse which is less likely to be useful to the user.

This manual way of validating the quality of search results can get very time-consuming and is only effective for comparing algorithms which have an observable gap in result quality.

Hence we have also tried to develop an automated heuristic for judging how good an output ranking really is. One solution we thought of is to make use of existing well-developed patent search technologies such as Google Patent Search. We can compare the top hundred or thousand results from our algorithm against the output from Google Patent Search and generate a score based on the number of matching patent documents present in both results list. Using patent IDs, finding the intersection of matching results would be simple and fast. However we found that since the input for our algorithm tends to be much longer than the typical number of search keywords, Google Patent Search tends to return only a handful of results or none at all for the same test queries. This is because Google treats all words in the paragraph as keywords and try find matching documents which contain all the keywords/phrases. This process eliminates potentially useful results such as in the case of synonyms where a relevant patent about cars might not come up in a query mentioning automobiles if the word is not present in the document. Our algorithm takes care of this problem by using word clustering when preprocessing the data.

Another idea is to collate search results from all our different algorithms and assign a score to each result in the list according to its position in the ranked list and how often and highly it is ranked in the result lists of the other algorithms. We then sum up the scores for the top hundred/thousand results, the algorithm with the higher sum should theoretically have better search relevance since its top results appear most frequently and get ranked highly across all algorithms. However at this point of time this is still a skeleton of an idea there are still many unknowns such as what values do we use for scoring and how to adjust them according to the position of a result in the ranked list etc.

We began this project to build a search engine for patent documents using a novel approach. While a lot of our initial time has been spent on setting up the data pipeline, designing and optimizing the search algorithm, evaluation of search results plays an increasingly critical part in helping us improve on our ideas. Building a reliable automated scoring function for results will go a long way in helping us to make incremental improvements to search quality, especially since we currently lack the main source of input that modern search engines rely on for learning: a large volume of user feedback.

Concluding Reflections

We started off the year with the goal to create a search engine for patents using novel document similarity techniques and while there is certainly room for improvement when it comes to optimizing our algorithm for speed and scale, I feel we have accomplished our initial goal of setting up the basic pipeline for patent search and the user infrastructure. We were also able to experiment with and evaluate many of the original ideas we had for the search algorithm and its optimizations, all the while working with and learning new technologies along the way. Of course, during the year we faced many technical challenges and limitations which made us re-evaluate our goals and scope for the project. It is these challenges that I would recommend for future work, especially with regards to performance speedup and search quality evaluation. Methods to run both the data preprocessing (BIDMach) and the lasso algorithm (ADMM) in parallel exists and more experiments can be done to test their performance on our problem. With access to machine clusters, many of our current challenges may be alleviated and the focus can be shifted to creating a suitable metric for evaluating the quality of search results.

I feel the method of agile software development we used in completing this project worked well for us, though it would be even more effective with shorter iteration cycles and better defined goals in each of them. It would be straightforward for future teams to pick up at where we left off as the code we used for retrieving and preparing the data, the algorithm we developed as well as the user interface are all well commented and accessible from our research group's code repository.

Acknowledgements

We would like to thank Prof. Laurent El Ghaoui for advising us during the span of our Capstone project. We would also like to thank StatNews research group that we are a part of for giving constructive advice and providing technical support.

Works Cited

Bosker, Bianca. Apple-Samsung Lawsuit: What You Need To Know About The Verdict. http://www.huffingtonpost.com/2012/08/24/apple-samsung-lawsuit-verdict_n_1829268.html, 2012: Huffington Post.

Google Patent Search. http://www.google.com/advanced_patent_search, 2015: Google Patents

High, Peter. Gartner: Top 10 Strategic Technology Trends For 2014 http://www.forbes.com/sites/peterhigh/2013/10/14/gartner-top-10-strategic-technology-trends-for-2014/, accessed February 15, 2015.

Masson, Patrick. Open Source Inititative: 2014 Annual Report. http://opensource.org/files/2014AnnualReport_0.pdf, 2014: OSI.

Page, L. Generic Method for node ranking in a linked database. http://www.google.com/patents/US6285999, 2001: Google Patents.

Carpenter, Brian; Hart, Judith; Miller, Jeannie
Jump starting the patent search process by using subject-oriented databases. http://www.engineeringvillage.com/search/doc/abstract.url?&pageType=quickSearch&searchtype=Quick&SEARCHID=152b761cM93aeM4669Mac30Me7750989c781&DOCINDEX=7&database=1&format=quickSearchAbstractFormat&tagscope=&displayPagination=yes, accessed February 11, 2015.

Carter, Brittany
IBISWorld Industry Report: Trademarks & Patent Lawyers & Attorneys in the U.S. http://clients1.ibisworld.com/reports/us/industry/default.aspx?entid=4809, accessed February 11, 2015.

Google Patent Search
2011 Advanced Patent Search. http://www.google.com/advanced_patent_search, accessed December 1, 2014.

High, Peter
Gartner: Top 10 Strategic Technology Trends For 2014
http://www.forbes.com/sites/peterhigh/2013/10/14/gartner-top-10-strategic-technology-trends-for-2014/, accessed February 15, 2015.

Costowl.com
http://subscriber.hoovers.com/H/industry360/overview.html?industryId=1063, accessed February 11, 2015.

How Much Does a Patent Lawyer Cost?
http://www.costowl.com/legal/lawyer-patent-cost.html, accessed March 11, 2015.

IBM

2014 IBM Research. http://www.research.ibm.com/careers/, accessed February 15, 2015.

Laurent El Ghaoui, Guan-Cheng Li, Viet-An Duong, Vu Pham, Ashok Srivastava, Kanishka Bhaduri.
2011 Sparse Machine Learning Methods for Understanding Large Text Corpora. Proc. Conference on Intelligent Data Understanding.

Lawyers.com
2014 Find a Patents Lawyer or Law Firm by State. http://www.lawyers.com/patents/find-law-firms-by-location/, accessed December 1, 2014.

McCarthy, Jerome E.
Basic Marketing. A Managerial Approach, 1964.

Mintel. (2014). Pharmaceuticals: The Consumer - U.S. - January 2014. http://academic.mintel.com/display/692963/, accessed February 11, 2015.

Mintel. (2014). Coffee - U.S. - August 2014. http://academic.mintel.com/display/713740/, accessed February 11, 2015

Nolo LAW for ALL (NOLO)
2014 Patent Searching Online. http://www.nolo.com/legal-encyclopedia/patent-searching-online-29551.html, accessed December 1, 2014.

Porter, Michael E. "The Five Competitive Forces That Shape Strategy." Special Issue on HBS Centennial. Harvard Business Review 86, no. 1 (January 2008): 78–93.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein
Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers
Foundations and Trends in Machine Learning, 3(1):1–122, 2011

Samuel Smith, Jae Yeon (Claire) Baek, Zhaoyi Kang, Dawn Song, Laurent El Ghaoui, and Mario Frank.
2012 Predicting congressional votes based on campaign finance data. In Proc. Int. Conf. on Machine Learning and Applications.

SBA
Small Business Trends, https://www.sba.gov/offices/headquarters/ocpl/resources/13493, accessed February 15, 2015.

United States Patent and Trademark Office (USPTO)
2014 Patents for Inventors: Patents - November 2014.
http://www.uspto.gov/inventors/patents.jsp, accessed December 1, 2014.

United States Patent and Trademark Office (USPTO)

2013 Glossary - September 2013. http://www.uspto.gov/main/glossary/, accessed December 1, 2014.

David M. Blei, Andrew Y. Ng, Michael I. Jordan.
Latent Dirichlet allocation. Journal of Machine Learning Research, 3, 993-1022, 2003.

Marcus A. Butavicius, Michael D. Lee.
An empirical evaluation of four data visualization techniques for displaying short news text similarities. International Journal of Human-Computer Studies, 65, pp. 931-944, 2007.

Xi Chen , Yanjun Qi , Bing Bai , Qihang Lin , Jaime G. Carbonell
Sparse Latent Semantic Analysis
SIAM International Conference on Data Mining (SDM), 2011.

Tibshirani, R.
1996 “Regression shrinkage and selection via the lasso.”
Journal of the Royal Statistical Society. Series B (Methodological), Volume 58, Issue 1, 267-288.