Paul M. Washburn
CSCI E-11 Term Paper
December 18, 2019

Office Networking Application

The *Office Networking App* that Dr. Brown utilized to "*see which relationships currently need attention*" (page 7) is the vignette that stood out to me as a technological innovation that I'd like to explore more deeply. This vignette covers all of the technologies discussed: big data, IoT, and cybersecurity.

The proposed application would capitalize on the current loss of productivity stemming from suboptimal interpersonal relationships between professionals that are supposed to be organized around a common goal. In the past I have assembled crude tools based on graphical analysis that are aimed at doing this precise task, motivated by a desire to manifest productively harmonious professional relationships with key players in my organization. This idea can be extended to meet broader markets in future iterations (should it see success in the professional domain). Future iterations of this idea could be extended to a social context with the objective of improving interpersonal relationships as they pertain to ones' own life goals and values.

The technology I'd like to explore most deeply in this application would be Big Data aspects of implementing an effective manifestation of this idea. Three aspects of this topic are relevant to the problem: (1) the integration of various of data sources that describe a user's interpersonal relationships, (2) the design of the data systems & storage schema of said data, and (3) the analysis of said data to produce machine learning algorithms and logic systems that are useful to the user.

Big Data

There are numerous technologies covered in this unit on Big Data that are likely to be crucial in implementing a successful Office Networking App. Each will be discussed in-turn below.

One of the most exciting technologies to be employed in the implementation of an office networking app will be effectively leveraging modern tools to wrangle & unite all of the disparate data sources relevant to the modern office environment. These tools will enable Dr. Brown to not have to worry about integrating her data by hand. Important examples of data that users like Dr. Brown may want to integrate include popular communication platforms such as Slack or Microsoft Office – for which integration may be automated via API. However, dealing with the "long tail" of alternative sources (such as information transcribed from audible conversations, physical fascmile, or company cell phone text messages) may be tougher to integrate — so for this a tool like Data Tamer might be leveraged. The same idea is present in mining the web for "business reputation" indicators. All this data will need to be integrated in order to be maximally useful to the Dr. Brown and her daily office interactions. Data can be integrated similarly to some of the applications discussed in class – via some data concatenation, normalization, and distance from other points (via cosine similarity or some other measure of distance). Getting the data into one centralized source will be essential to the success of this application.

Cloud technologies of all breeds will be a cornerstone in the success of the proposed venture, especially as they relate to storage/architecture and analytics. The cloud will enable quick setup, outsourced system administration, elasticity & reliability of the system, and quick setup of scalable compute. A columnar storage model will be preferred for all database storage to enable efficient retrieval and storage space. An ArrayDB-like system will be employed for efficient ML-OPS, such that a query and a complex analytical operation (such as Singular Value Decomposition) can be performed in one job. This will be vital for supporting tens of thousands of personalized ML models for users like Dr. Brown. Dr. Brown may be interested in finding a conversation from the past, thus invoking a search operation that is "embarrassingly parallel." For straight-forward & parallelizable jobs like this, Hadoop will be employed such that the work is distributed over many workers via a map-reduce operation. Tasks such as exploratory analysis and large-scale training of ML models will be performed using Spark to take advantage of main memory computations. The same memory-first approach will be taken with the firm's OLTP database for housing the firm's transactions.

Some of the data users like Dr. Brown will value from this application will be of high velocity, and in order to make timely use of it there will need to be some sort of approximation in computation & simplification in storage. For example, analyzing graph data of Dr. Brown's calendar appointments with colleagues will contain a lot of noise (e.g. any given meeting typically won't amount to much). However, identifying higher order patterns by incorporating data from users unrelated to Dr. Brown may help shed light on the health of her inter-colleague relationships. This invites the use of coresets for representing data. This will allow for minimal error in computations at the benefit of timely answering of important analytical questions. Some data of this sort will be dealt with via streaming (summarized yet fully evaluated) while some will be dealt with via sampling when outliers are not prevalent.

Investigating data for intuitive understanding is essential for actionable business understanding. That is why interactive data visualization will be allocated more resources for this office networking app than is typical in similar organizations, giving Dr. Brown and other users a sort of Command Center Dashboard for office interactions. Allowing users like Dr. Brown to perform direct manipulation on their data via a user interface that centers upon a visualization of said-data helps facilitate a better understanding of the nuances of office networks. Dr. Brown will be able to pan, zoom, filter and facet the data in real-time to investigate hypotheses and check on the health of relationships at work. Of course, visualizations will be designed to have an appropriate graphical integrity ratio (near 1) so Dr. Brown is not shown more information than is necessary to communicate the information sought.

Machine learning of all sorts will be central to this application to enhance user experience and knowledge. Plain vanilla classification systems will be used where appropriate (and supported where appropriate with big data tools), yet so will regression and NLP systems. Multi-aspect summarization is another interesting technology that can be applied to the proposed Office Networking App and will be explored in an R&D context for this task. Unsupervised labeling of Dr. Brown's digital/ typed sentences (whether from Slack or from an in-person meeting) might help a her identify meaningful conversations & changing relationships, such as when a conflict may have arisen without her knowledge. Implementing recommendation systems will also be enabled by implementing co-occurrence matrices for users vs. features, and inform recommendations for actions to improve their relationships & outcomes at work. Property testing will be helpful in the acquisition of a representative sample for use in training machine learning models when there is so much potential data that the noise-to-signal ratio is an impediment.

As the amount of data Dr. Brown generates grows, so too will the data generated per user, and it will continue to grow at an exponential pace. The big data technologies covered in this module will be of ever-increasing importance to the successful implementation of the proposed office networking app. It is likely that more research into related technologies will bear fruit as well.

Internet of Things - IoT

Several technologies from the IoT unit could be combined in the development of an Office Networking App. Unlike many industrial applications of IoT technologies – which are typically straightforward collections of data streams relevant to some production process – this use-case requires creative applications of IoT to help deliver value to professional users like Dr. Brown.

One application IoT is the use of networked microphones to collect a verbalized journal entry of each user that describes their professional interactions for the day. This user-centric human-computer interface will enable the creation of data that can be mined for patterns by the Office Networking App. A portion of these microphones could be networked from existing applications, such as a self-driving vehicle, so that users like Dr. Brown can either be commuting or in their office when they dictate their diary entry for the workday. Dr. Brown's location when he records his dictated journal entry will determine how the data processing & transfer will take place. When he is in a community self-driving vehicle, existing on-board microphones are powered by the vehicle's main computer system and thus will not have battery concerns. Neither data volume-per-day nor device-to-gateway range will be a concern in this scenario since the vehicle can be used as a "data mule" and transmit Dr. Brown's data when connected to WiFi while it charges at night time. Microphones will need to be installed in his office to allow for times when Dr. Brown is dictating his thoughts about his daily professional interactions when he is physically in the office. By installing a few microphones in his office that are directly connected to power, Dr. Brown can choose whether to dictate his thoughts when he's in office or during his ride home – both of which are private environments. If the entry is recorded in the office then the office WiFi can be used to transmit data to the cloud; otherwise the community self-driving vehicle will prompt a data dump when it is idle and connected to WiFi for the day. Device-Cloud architecture is used here since there are no "real time" needs placed on the system, and thus no need for device-local or fog computing.

Indoor localization techniques combined with inexpensive indoor temperature sensors will be leveraged in this application to ensure Dr. Brown and his colleagues are never too-hot or too-cold in a meeting again – ensuring everyone is physically comfortable and ready to contribute. Integrating each user's temperature preference data with meeting attendance data from the Virtual Assistant will inform the Local Warming technology which users are comfortable at which temperatures. Users like Dr. Brown will be able to initialize their desired temperatures as a starting point, and over time the system will use wireless reflections of the human body to identify if a given individual has goosebumps (indicating they are cold at the current temperature setting) or if they are sweating (indicating they are hot at the current temperature setting). If a too-hot or too-cold flag is raised during a meeting, then at the end of the day the Office Networking App can ask Dr. Brown a quick *"Were you too hot/cold during your noon meeting today?"* to provide feedback into Dr. Brown's preferences. By being cognizant of users' temperature preferences we can ensure that Dr. Brown and all of his professional colleagues using the application will always be physically comfortable during meetings. The days of the freezing or sauna-like meeting rooms will be over.

The above indoor localization techniques could also be combined with camera data & wearable data to direct the lighting in a given meeting room at exhibitions focused upon by the speaker's inferred angle of gaze. While highlighting the object of Dr. Brown's presentation, this system will also help Dr. Brown's audience take notes by shining lighting directly on their notebooks as they write. When a given member of the audience is not taking notes, their lights will shut off to save on energy.

The microphones and cameras installed in meeting rooms will also be used to identify and surface "tense" interactions between colleagues, effectively identifying negative outliers in the productivity of a professional interaction. This will be especially useful in large meetings where interactions between colleagues are rapid-fire and pass by quickly. Voice recognition, speech to text, sentiment analysis, facial recognition & micro-expression classification (among other machine learning techniques) can be applied to the data collected in order to identify such "tense" interactions. Furthermore, indoor localization techniques similar to those described above could also be used to measure breathing and heart rates to identify large jumps in heartrate and/or breath, thus augmenting the feature set. Anomaly detection similar to what was discussed in lecture could be used to identify commonalities among outlier events. Since this is an example of outlier detection, much of the raw data describing "non-tense" interactions will not be of much interest. In the spirit of keeping only the data we need, raw data from neutral interactions will be discarded, keeping only the metadata and outputs of the summarization processes for these records. For interactions that were sufficiently "tense," all of the raw data (e.g. audio/video, text translation) will be kept alongside the summarization process to allow for deeper analysis. While hopefully the vast majority of interactions that Dr. Brown has with his professional colleagues are not "tense", this feature will be able to notify him when it occurs – and recommend some actions to help mitigate the damage & improve the relationship moving forward.

Finally, the existing cameras in both meeting rooms and common dining areas can be used to implement facial recognition combined with object detection to gather data on which colleagues enjoy which foods that are arrayed in the common area. This will enable the collection of dietary preferences that users like Dr. Brown can employ when deciding where they'd like to take a colleague to lunch, or what sort of treat would maximally delight a given colleague.

There are few straightforward approaches to using IoT in an Office Networking App. By configuring a minimal IoT setup we are able to build value in the proposed Office Networking App by improving the human-computer interface and by maintaining an optimal physical & emotional work environment.

Cybersecurity

To keep Dr. Brown's data safe, this Office Networking App will employ elaborations of three defensive strategies: prevention, resilience under attack, and incident detection & recovery. A holistic view of security will be employed to implement a robust systems-wide security architecture so we can ensure we've taken all steps that are feasible and prudent towards the goals of protecting Dr. Brown's, as well as other users', data confidentiality, integrity and availability.

Given we won't know in advance who might attack, or what they might do, we need a threat model that is conservative yet realistic. Since cybersecurity is truly a property of our computer system as a whole, a holistic threat model would not only include software vulnerabilities. Our threat model will also take into account physical attacks & social engineering attacks. We will assume that an adversary that poses a threat to the Office Networking App's security will be capable of controlling some (not all) of the network's computers, controlling some (not all) of the software on these machines, as well as identifying & exploiting relevant bugs in our software base that may pose a security threat to our ecosystem. We assume an adversary will be capable of stealing secret keys, using public keys alongside them (and using them to reverse-engineer ciphertexts), and exploit system protocol. Furthermore, we'll also assume that adversaries might try to use social engineering attacks to gain access to individual or administrator user accounts via deception and/or outright theft.

There are numerous ways that we can *prevent* certain known attack vectors. Some standards such as the use of securely typed programming languages and employing public-private key cryptography will be bread-and-butter approaches, yet other prevention mechanisms will be instituted as well. To avoid code injection via string fields in the application, data sanitization will be employed for server-side tasks that typically concatenate strings from a user. In order to avoid confused deputies accidentally leaking data to unauthorized users, security labels will follow data to verify whether it can be exposed to other users via the privacy policy. (Resin on Rails will be the web framework we'll use to implement this feature.) Server-side, this app will consider a new app architecture that is Android-like in its concepts in that we will seek to implement isolation between applications & services (e.g. User databases will be separated from Payment databases, and all functionality will be isolated). Randomized disk-block encryption can also be used to avoid system overheads, keep file encryption design simple across the organization, and ensure the same encryption system works on different file systems — keeping Dr. Brown's files safe even if an adversary with knowledge of our cryptography tries to plop down a different operating system during a breach. A Trusted Platform Module can be used to implement measured boot — preventing adversaries from accessing server applications without an exact *TPM_extend* match. The final prevention mechanism that we'll use server-side, which will help with the Trusted Platform Module, is Microsoft's BitLocker. On the user-side, this application will employ built-in biometric authentication mechanisms will be combined with signed boot procedures to augment the application’s authentication protocol. Finally, we will protect web users by employing HTTPS (so we can encrypt communication with users like Dr. Brown) as well as identify “trusted” regions of the network through which our traffic can flow. Prevention of these common attack vectors will help Dr. Brown and other users of the app trust that what they tell the Office Networking App, as highlighted in the vignette, will remain confidential.

In order to remain *resilient under attack*, the Trusted Computing Base will be shrunk as much as possible. One obvious way of making progress towards this end would be deployment of tamper resistant hardware since only portions of such hardware are “trusted”. Ascend is a good option for such hardware given that it obfuscates pin traffic from an adversary’s view. Another mechanism that will help us to keep Dr. Brown’s data safe is privilege separation, splitting functionality onto several machines or VMs, which will help compartmentalize breaches and thus isolate the damage that can be inflicted by any one unauthorized breach. Privilege separation will also help us to quarantine subcomponents & systems if and when they are penetrated, thus minimizing damages to Dr. Brown.

As for the *detection & recovery* strategy, it is important to understand that compromises will be inevitable; we’ll need both proactive and reactive strategies to mitigate damages to users like Dr. Brown as well (as to the organization) if and when we are breached. For recovery, a tool like Retro will be on-hand to help disentangle legitimate changes made by authorized users like Dr. Brown and those made by an adversary. This will minimize the input necessary from Dr. Brown to recover her information should some of her data be affected by a breach. Assuming an adversary will eventually gain control of a subset of our system, performing system roll-backs & re-executions will need to be performed such that we preserve as much legitimate user data as possible. This will be vital to keeping Dr. Brown and other users happy with our services.

The fact that there exist many untrustworthy network operators and users renders the Internet a somewhat insecure medium, and taking a straightforward & holistic approach to cybersecurity will help raise barriers to attack while also remaining resilient when one occurs. While complete security is impossible we still must make a best effort to protect users like Dr. Brown.

In Depth Review: Secure Multi-Party Computation

This Office Networking App will be privy to large amounts of data that, if leaked, might cause real damages to professionals like Dr. Brown that utilize it. However, there will be many administrators (such as employers, public & private) that will want to benefit from the collective knowledge that might be gained by gathering together all of the collective data generated using the application and analyzing it in various ways. We'll discuss secure multi-party computation and how it will be applied to the app to deliver more value without sacrificing security.

Classical cryptography addresses authenticity and privacy of communication, or messages sent in plain text. As the velocity of data increases in society, however, there is more demand for a new goal: to compute arbitrary functions using data owned by many users — without divulging un-owned data to any player. Often in these situations there will exist subsets of colluding players that are curious to learn about data that is owned by another party. These colluding players may be malicious or honest & curious; either way we will need a mechanism to keep data private. An obvious but perhaps naive approach might be to implement a "trusted center" solution to perform computation on the combined dataset, but this approach leaves us exposed to the potentiality that the center node might be faulty and act in malicious ways.

Here is where secure multi-party computation enters the scene. Secure multi-party computation enables us to compute via a decentralized protocol that emulate a "trusted center" solution while ensuring correctness, privacy and independence of inputs into the global function. A key tool to successfully implementing a secure multi-party computation scheme is secret sharing among all of the parties. In this protocol the secret *S* must have three properties: each party receives only a share of the secret *S*, no player can recover the secret on their own, and if all players act together then they will able to recover the secret.

Sum sharing is the mechanism that allows us to achieve a secret with these properties. Sum sharing includes choosing a prime number *P* that is greater than the secret *S*, then generating a vector of random numbers between 1 and the prime number *P* that is of length N-1 (where N is the number of parties). The final value in the vector will then be set to summation of this vector subtracted from the secret *S*, modulo the prime number *P*. Each player is then distributed their allocation of the random vector. The dealer then splits the secret *S* and distributes share of this vector. Since the all values in the random vector (except the final value) were chosen independently of the secret *S* they contain no information about *S*. Only when you have the last value can you extract *S*.

Via the Completeness Theorems we are ensured of the following. First, if there is an honest majority of players (less than half are faulty), then any multiparty function can be computed securely — even when there are colluding parties that are curious but non-deviating. Second, assuming an honest 2/3 majority, then any multi-party function can be securely computed even when colluding players maliciously deviate from the secure protocol. Finally, assuming an honest majority and an ability to broadcast, any multi-party function can be computed securely even if faulty players deviate. This provides *unconditional security* — meaning no complexity assumptions were made on the power of the adversary.

Secure multi-party computation techniques will be leveraged in this Office Networking App to distribute trust amongst different organizations that are utilizing the platform. Program administrators of this Office Networking App will be hesitant to share their data, despite the fact that sharing data across organizations will enable them to learn more about the workforce via the signal present in other organizations' data. By leveraging this protocol we will be able to perform arbitrary functions on the combined super-dataset while enjoying the unconditional security the protocol provides.