GDPR DISCOVERY USING TEXT MINING AND DEEP LEARNING #23120

Elinar Oy Ltd
Ari Juntunen
Chief Technology Officer
AGENDA

• GDPR - what is the problem
• Architecture
  • Solution components
  • Watson APIs
  • Networks
• Performance
• QA
GDPR – NEW DATA PROTECTION REGULATION

- GDPR requires companies to design their systems using “Privacy by Design”
- Companies storing privacy data of EU Citizens must have their consent to do so
- Data subjects have significant rights, including:
  - Right to access their data
  - Right to get copy of their data (Data portability)
  - Right to be forgotten

- Is extra-territorial; applies to EU Citizens worldwide..so US based companies must adhere or face severe penalties (up-to 4% of global turnover)
WHAT IS PRIVACY DATA?

• Privacy data is very wide; anything that can identify person is Privacy Data
  • Regulator in Finland has made a decision that mileage of a car is privacy data
• It depends on context; a piece of data could be privacy data in one context but not in another
• Name is not (necessary) privacy data but when combined with other information like birth date or address
• Some privacy data is easy to detect like Social Security Number or Customer Number
SIMPLE APPROACHES WORK FOR SIMPLE USE CASES

• There are number of good solutions when privacy data is simple
  • For example IBM StoredIQ; if it can be expressed using RegExp, StoredIQ can extract privacy data with astonishing speed
• Unfortunately not all privacy records are that simple
• Privacy information is sometimes spread out and they still make a record as a whole
RIGHT TO ACCESS / RIGHT TO BE FORGOTTEN

• GDPR becomes in effect May 2018
• From that point on all registered subjects may request to see (and to get) all data that company has stored about them
  • This is not even limited to electronic material, but applies to paper archives and even to tombstones
• In large corporation it is not practical to run extensive discovery process every time a person wishes to see everything that is stored
  • A one time discovery process needs to be run on
  • This is where Deep Learning will come in
USING DEEP LEARNING TO DETECT GDPR DATA

• Deep Learning is quite obvious solution for detecting GDPR data
• Solution needs to be highly scalable
  • Large corporations might have hundreds of billions of candidates for GDPR data
• Solution needs to be able to understand unique data structures for each corporation
• There must be easy-to-use interface to provide and create training data
AGENDA

• GDPR - what is the problem
• Architecture
  • Solution components
  • Watson APIs
  • Networks
• Performance
• QA
ELINAR GDPR AI MINER WITH IBM WATSON

IBM Bluemix / Watson Cloud

Elinar GDPR AI

Customer on-prem system
TECHNOLOGY BASE

- IBM Watson Document Conversion Service creates a text representation of data record or document
- IBM Watson Document Classification determines Geo – area that document refers to
  - This is important so that we can use correct formatting for ZIP codes for example
- IBM Watson NLU is used to create Features for each word in a text
  - Features tell Neural Network if it is Name, Phone Number, Street Name and so on
- IBM Power8 “Minsky” with direct NVLink connection from CPUs to P100 GPUS used in development
- Sdfs
IBM POWER8 “MINSKY” & POWER.AI

• Very powerful package, 1 TB memory, 2xPower8 CPU, 4XP100 GPU, 2 GPUs connected to each Power8 CPUs directly using NVLink
  • NVLink between GPUs connected to same CPU
  • Ubuntu 16.04
• Power.AI: Power8 optimized DL frameworks, including Tensorflow, Torch and Caffe
• Power8 optimizations very good
  • On x86 we have to use Lua instead of LuaJit due “Luajit out of memory” when creating larger models. This is non-existent issue in Power.AI
• Minsky does ~284 Mhash/second on Ethereum Mining :P
Elinar GDPR AI MINER

Elinar REST

GDPR data
Yes/ No

Neural network

AI GDPR

Bluemix
0.02 c/ query?

Watson APIs

- Document Conversion
- Natural Language Classifier
- Natural Language Understanding
DATA PREPARATION

• Business Data can be very long
  • For practical purposes a sliding window approach must be used
• In general GDPR data is found within certain distance from each other, this makes sliding window approach very feasible
  • Default sliding window was determined to be ~300 words -> This is approximately one page on a business document
• Must use overlapping windows
• Each token is annotated with a type (For example Ari is Name and Juntunen is Last Name)
DATA PREPARATION

- Figures about Watson NLU here.
ISGDPR AI

- Simple Single layer 500 neuron network using Torch
- Provides two outputs:
  - True
  - False
- Very high throughput
  - Numbers go here!
- High confidence with relatively small training set
  - Fast to train
- Experimented quite a lot with more “advanced” architectures like 1000x4 and so on
  - This kind of classification problem does not seem to benefit from larger networks when sliding window is 200 or 300 words
- Word and Feature Vector Sizes do matter a lot
Training mode

**Phase 1**
- **Training mode**
  - **Training Module** (Taught by the customer)
  - **Pre Made AI**
  - Custom defines if data GDPR or not.

**Phase 2**
- **Validation API**: Customer teaching, time delay
- **Configuration**
  - **Annotation NLU (Watson)**
  - **AQL** (BigInsights)

**Phase 3**
- **Rules, defined by client, when document has GDPR data, AI needed?**
- **EXTRACTOR AI**
- **AI GROUPER**
- **RULES**
- **EXPORTER**
- **Configuration**
  - Name, date of birth, address
  - Name, address
  - Social security number, customer number
  - Level 1
  - Level 2
  - Level 3

**Natural Language Classifier**

**IS GDPR AI** (Caffe2)

**IBM Watson**
- **Document conversion**
- **Natural Language Classifier**
- **Natural Language understanding**

**Validation Loop**
- Feedback loop
- YES
- NO
GDPR EXTRACTION PROCESS

• Our approach is dictionary based
• But had to resort to fairly unique “Typed” UNK token handling so that unknown tokens can be reliably mapped from document to results
  • UNK Tokens like names and dates are the actual records values
  • Attention based models did not perform well enough (we must map every item in record correctly)
• Each type of UNK-token (like date or social security number) have their own special array
  • 1st UNK of a type will be #1 on this array and so on
  • AI will have to learn to pick up the structure and layout of records from surrounding text
ANONYMIZATION

- Since we map all uncommon tokens (like names, addresses and ZIP codes) into special UNK arrays actual values that are contained within them are irrelevant
  - They have to be consistent within a document
- This means that we can replace a specific token like name “Ari” with anything that our pre-processor (Watson NLU) recognizes as name entity
  - All instances of particular entity within a document must be replaced with same replacement entity
- This way we can anonymize training data, subject matter experts (like HR) who prepare training data for their department work with real data but as material is saved into training set all personal information is replaced with common tokens
  - This is great for troubleshooting as well, devs learn to read this data quick
TRAINING DATA GENERATION

- This approach requires significant training material volume
- Some can be acquired from IT systems, but commonly some need to be prepared by manual labor
  - → Need for streamlined user experience
- Once manual material is generated there is a way to increase set size by using simple strategy that seems to benefit LSTM:
  - Create a copy (or several) of training set with 2-12 random tokens added into beginning of each sample
  - → This moves records on the documents slightly and enables AI to pick them up with higher accuracy with a cost of 12 extra tokens / document
NETWORK STRUCTURE

• Suprisingly shallow topologies seem to work well
  • Deeper topologies seem to be more suitable if large number of samples are available

• Some findings with our UNK –tokenization based method (encoder and decoder are equal) using LSTM:
  • 1000x1 ->Useless
  • 1000x2 ->Very good for small training sets, very good performance
  • 1000x3 ->Gives errors with small sets like extra tokens pointing to dictionary entries
  • 1000x4 ->Not much benefit

• Word vector length does have surprising effect:
  • Longer the better. Max sentence length 320->good results on 2000-3000 range
AGENDA

• GDPR - what is the problem
• Architecture
  • Solution components
  • Watson APIs
  • Networks
• Performance
• QA
PERFORMANCE

• Watson NLU is the slowest API, each call takes ~500 ms
  • But it can scale parallel in massive way, for example 10 concurrent calls take also ~500 ms
  • Must use threads to keep AI busy

• Rest of the performance figures go here!
AGENDA

• GDPR - what is the problem
• Architecture
  • Solution components
  • Watson APIs
  • Networks
• Performance
• QA
CONTACT US

CTO Ari Juntunen
+35840 524 4482 | ari.juntunen@elinar.com

www.elinar.com

@ari_juntunen
@EinarOyLtd

linkedin.com/in/ari-juntunen-38637464
https://www.linkedin.com/company/2622808/
ELINAR
enjoyable solutions
since 1994