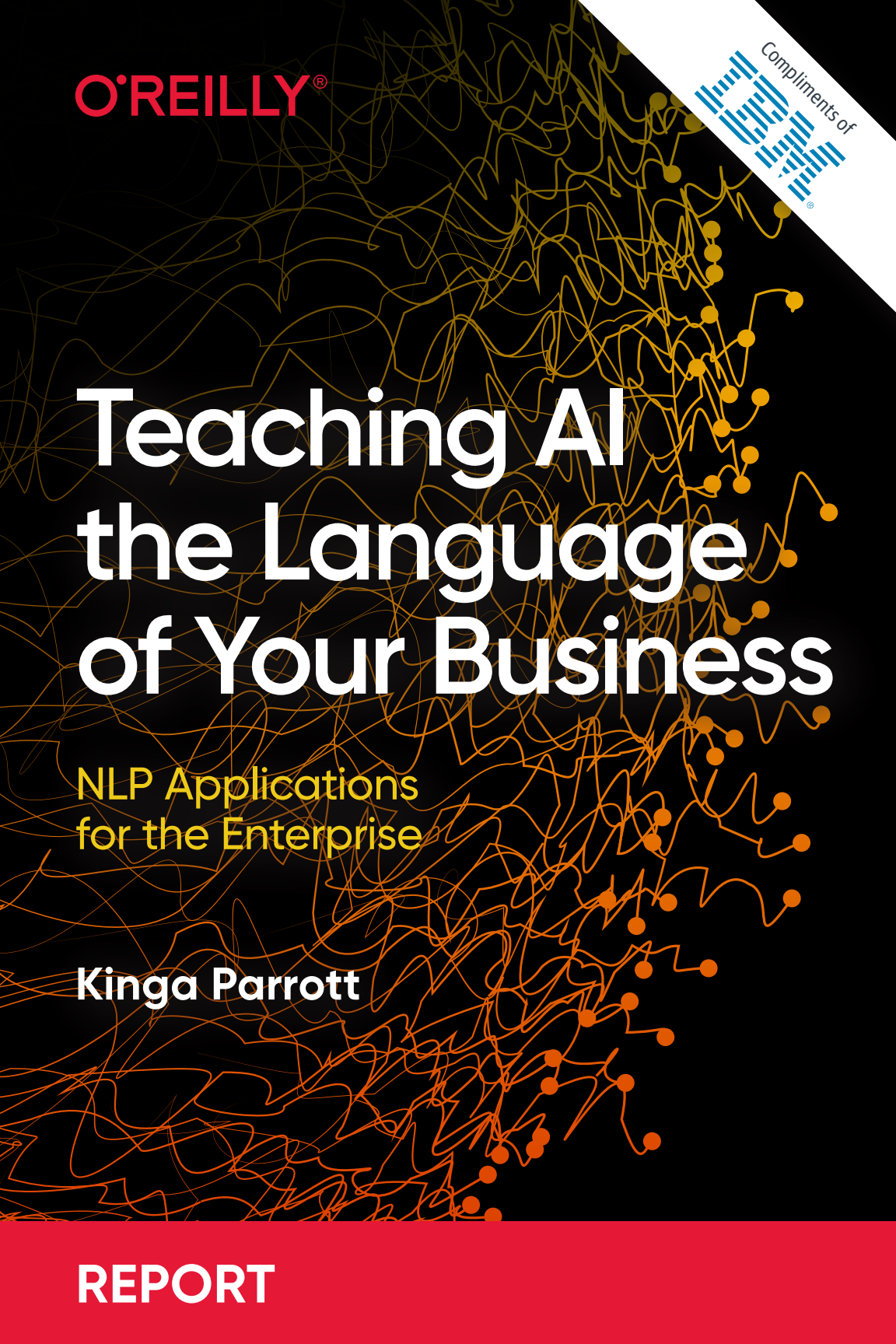
O'REILLY®
Compliments of
IBM
Teaching AI the Language of Your Business
NLP Applications for the Enterprise
Kinga Parrott
REPORT

Teaching AI the Language of Your Business

NLP Applications for the Enterprise

Kinga Parrott

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

Teaching AI the Language of Your Business
by Kinga Parrott

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Acquisitions Editor: Rebecca Novack
Development Editor: Nicole Taché
Production Editor: Caitlin Ghegan
Copyeditor: nSight, Inc.
Proofreader: Paula L. Fleming
Interior Designer: David Futato
Cover Designer: Karen Montgomery
Illustrator: Kate Dullea

October 2021: First Edition

Revision History for the First Edition
2021-10-27: First Release

978-1-098-10861-8

[LSI]

Table of Contents

Teaching AI the Language of Your Business

Most of us interact with some form of AI on a daily basis, both at home and at work. Every time we do a Google search, use Siri, book a ticket, get a Netflix recommendation, or receive a credit card fraud alert, we are interacting with AI-enabled applications. It is also often an AI-enabled function that decides to approve or deny a social service or a credit application, to reject a job application, and to file an email as spam. With faster computers and better algorithms, more and more tasks are being automated, increasing productivity and efficiency, and one area that has seen great advancements in the past couple of decades is natural language processing (NLP). (See "Basic Concepts" on page 3 for a definition and examples.)

NLP helps machines understand human language. The ability to process natural language has given rise to voice-based assistants, machine translation services, analysis of social media for product management, intelligent content management and extraction, and many other industry applications. But language is hard for computers to understand. The language we use to communicate is not just a collection of symbols—it is rich in context beyond mere words and makes it very difficult for machines to interpret. As luck would have it, much of the information that organizations can get insights from or use to automate functions (such as product reviews in social media posts, customer call center recordings, physicians' notes, contracts, and résumés) is natural language stored in the form of audio, text, images, or a combination of the three.

Introduction

As we will see in the use cases in this report, the first attempt to automate a process or extract insights from data in natural language often leaves businesses disappointed with the results. Even companies with well-established operations in AI can face this disappointment. For more on operationalizing AI, I suggest the O'Reilly Radar report *Operationalizing AI*, by Paco Nathan, William Roberts, and John Thomas.

As it happens, even with vast amounts of data, machine learning (ML) models built to interpret language need pointers and labels in order to understand which words and relationships between words are relevant or meaningful to the task at hand. On top of understanding the words and relationships between them, ML models must also understand how humans actually think about the data and/or processes. The process of accurately assigning these labels is called *annotation*.

Generally, during the annotation process, humans transfer their intelligence (domain expertise, cultural references, language, feelings, etc.) to an AI. Therefore, the more expert behavior we want an AI to display, the more important it is for the annotations to be accurate. And annotation accuracy requires the dedication of subject matter experts (SMEs), time, and a well-thought-out process. Who is the best person to train the machine to think about a topic like a human if not the domain expert? Who can best teach a machine how to interact with a customer if not the customer service representative? Teach the computer what information is relevant so it understands the risk associated with a contract? To teach the computer to hold a conversation? A conversation specialist?

In this report, we will show that to reduce the time to value of an AI project involving natural language, the project team needs to include experts in the relevant field and bring them in early. Furthermore, collaboration between humans and machines is a continuous loop—for AI systems to continue to produce results for the business, both must learn from each other.

Many texts refer to this process of human–machine collaboration as the *human in the loop* (HITL). I refer to it in this report as "teaching AI the language of your business," and my hope is that, by reading this report, more organizations will understand and prepare for

annotation and subject matter expert involvement. The result will be a quicker realization of the benefits of NLP.

The use cases presented in this report have been organized in order from lowest to highest *complexity*, complexity being defined as the number of elements required in the workflow to get to the desired result. Some use cases are more business oriented while others include more technical details; don't let this discourage you if you were hoping for a more (or less) technical report. My hope is that at the end, you will come away with some tools to prepare your organization for an NLP project, whether you are on the technical team or on the business side.

Before diving into our first use case, see the below summary of three "Basic Concepts" that are featured in this report—machine learning, NLP, and text annotation.

Basic Concepts

Machine Learning

Machine learning is a subset of AI. It was defined in the 1950s by AI pioneer Arthur Samuel as "the field of study that gives computers the ability to learn without explicitly being programmed."

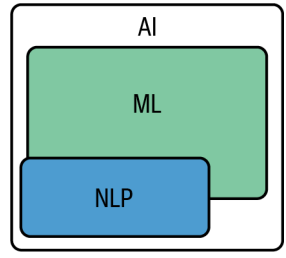

Figure 1. How AI, ML, and NLP relate to one another

It starts with the use of data and algorithms to imitate the way humans learn. An algorithm is a mathematical procedure/approach, such as the equation that represents a line y=a+bx. A model helps us identify the values of a and b, when the algorithm is provided enough pairs of (x,y). Those pairs are called the training set. Training is the process of giving a model inputs (features) and the ground truth (outcomes), and letting it learn how to *predict outcomes from features*. The ground truth is built using annotations, as

we'll see below. So, when you run an algorithm over your training data, the model is making predictions on new data.

Natural Language Processing

NLP refers to the branch of computer science—and more specifically, the branch of artificial intelligence, or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can.

According to IBM, "NLP combines computational linguistics—rule-based modeling of human language—with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to "understand" its full meaning, complete with the speaker or writer's intent and sentiment."

Major areas of application include:

Question answering systems (QAS)
: These systems answer questions posed by humans in natural language, leveraging information extraction from multiple knowledge bases. Alexa and Siri are simplified versions of a QAS.

Summarization
: This area includes applications that can generate a summary of the content of a collection of documents. Imagine being able to summarize a long book or, more interestingly, a collection of responses to an HR survey to understand the key topics of employee concern.

Machine translation
: A major area of research in the field, machine translation converts a piece of text from one language to another. Google Translate is a good example of this kind of application.

Document classification
: The task here is to identify in which category or class a document should be placed. This is one of the most successful areas of NLP, as well as one of the most used in enterprises.

Speech recognition
: Recognition of spoken language utterances is one of the most difficult problems in NLP. Conversational agents and Siri and Alexa use speech recognition, but more interesting applications can be found in customer service areas, where complex

interaction between the assistant and a customer takes place to solve an issue.

Text Annotation

Text annotation is the process by which meaning is assigned to a block of text, whether a word, phrase, or sentence. During annotation, pieces of text are identified and assigned a previously agreed-on label to provide meaning or context, as shown in Figure 2.

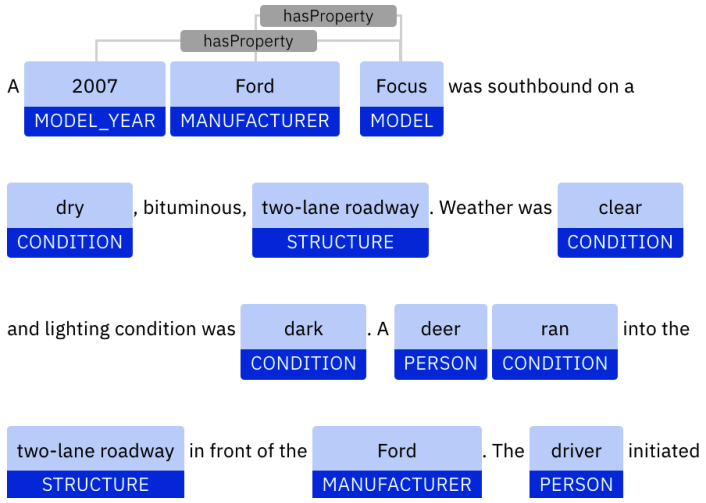

Figure 2. Text annotation

Annotations are always started by humans. However, there are ML annotators on the market that are basically models that have been trained with huge data sets to label text, enrich text, and perform other annotation tasks.

To understand the text annotation process better, it will be helpful to look at a couple of different types of text annotation.

In *named entity tagging* (NET) or *named entity recognition* (NER), the annotator is shown a block of text and possible tags. The annotator will assign relevant tags to each named entity, such as a city, a company, or a person. Alternatively, the annotation task may involve identifying a noun or verb or other part of speech (POS) or key phrases.

Sentiment annotation or tagging assigns a label that qualifies a word as having a positive, negative, or neutral sentiment.

In *semantic annotation*, the annotator will provide additional information to the text, such as domain-specific definitions and relationships between entities that may not be present in the document.

For more on text annotation, read *Natural Language Annotation for Machine Learning* by James Pustejovsky and Amber Stubbs (O'Reilly).

Use Case 1: Human Resources Document Management

HR departments are responsible for handling a multitude of documents with highly sensitive information. They struggle to manage documents coming from multiple sources at different times: from résumés, to contracts, to evaluations, to terminations, which can come as scanned documents, email, fax, web forms, and file downloads and even from other HR systems. Employee records are often spread across different systems and locations, making it difficult for HR to have a single view of any one employee. The larger the organization, the more complex the issue becomes.

Business Need: Capturing, Managing, and Protecting HR Documentation

An HR department must comply with an astounding number of rules and regulations. It must manage, store, and retain employee records properly and consistently, according to requirements. This means HR staff must know what documents they have, where they are, which ones they must keep and for how long, and who can access them. To maintain a centralized view of every employee, HR must capture and collect every document it receives and categorize it for retention and user access. To do this, HR must identify the type of document and to whom it belongs, archive it in the right file, and associate the right access authorizations.

At this global industrial technology company, there are more than 50 different records per employee, which means the HR department must handle hundreds and sometimes thousands of documents every day.

One of the first steps organizations can take is to automate the process, using robotics process automation (RPA), but even then, a

large part of the problem remains, requiring manual intervention: PDFs and scanned images as well as a large number of similar documents with different structures are hard for a computer to process (see sample documents in Figure 3).

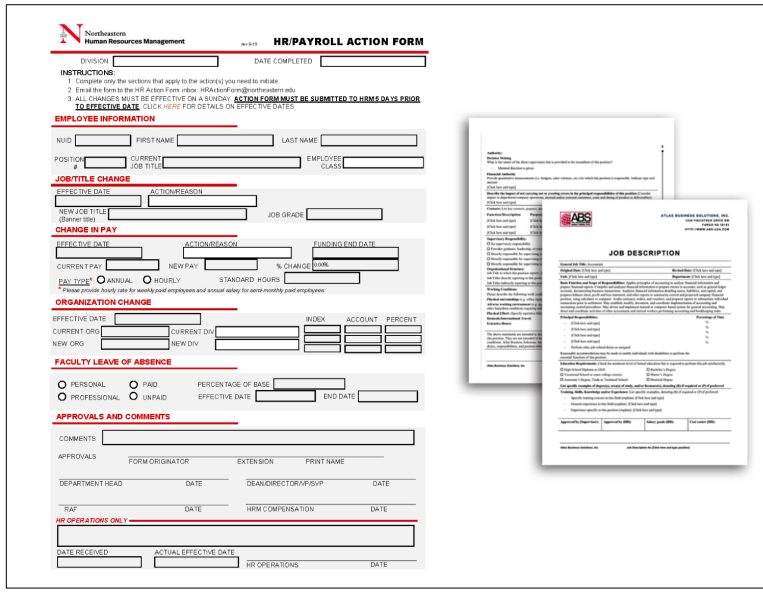

Figure 3. Sample HR documents

Solving the Problem: Systems Architecture, Technologies, and Techniques Applied

Let's walk through the general architecture of the solution, as illustrated in Figure 4.

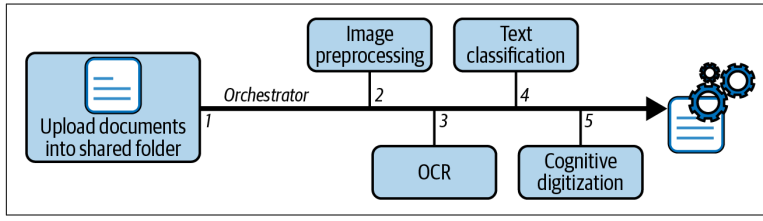

Figure 4. The general workflow includes image preprocessing and OCR steps to extract language

First, an employee scans or uploads the documents into the system (1). As we've mentioned, documents are stored and shared in

different formats. To classify the documents and archive them, we need to process the text. To process the text, we first need to extract the text from the tables and/or images with high fidelity, so the document goes through the image preprocessing module (2) to correct the orientation (if needed) and to eliminate noise (like shades, smudges, blurs) and images such as logos and lines (see Figure 5).

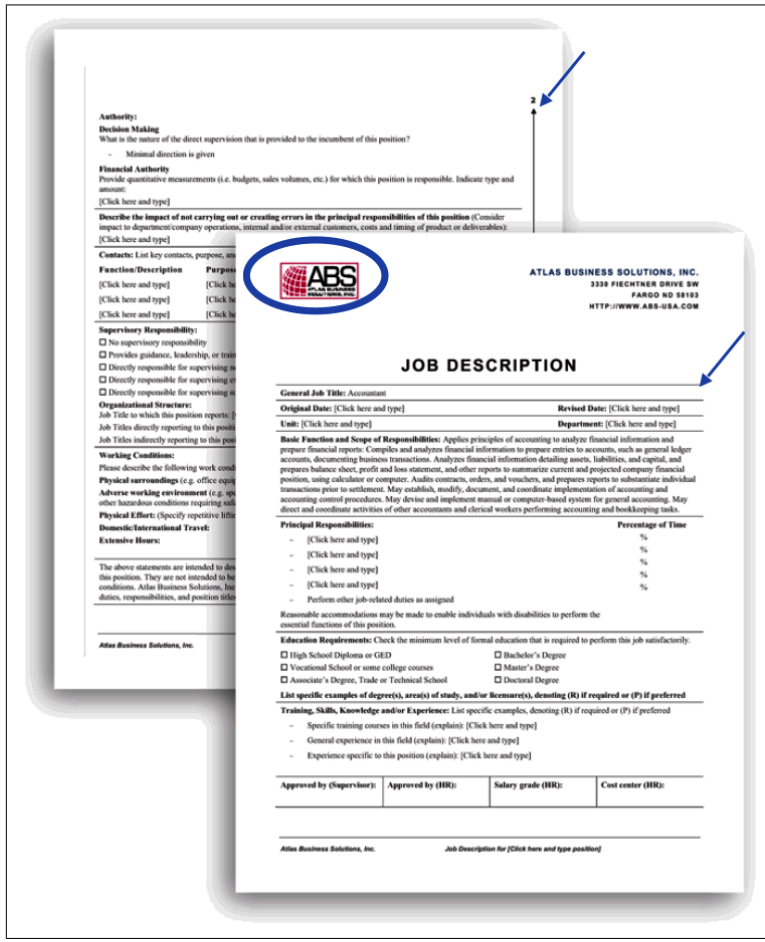

Figure 5. Example of the elements removed in the preprocessing module

When only text is left in the image within a document, it is sent to a deep learning–based optical character recognition (OCR) module (3). Here, text that is still stored as an image is recognized and digitized so it can be processed by a computer. This step is important,

because fidelity in the recognition of text is critical for an accurate outcome to the process.

In the next step, natural language understanding (NLU) and text-processing techniques are applied to classify the documents in the classification module (4). A custom model is developed to classify the documents. This mood currently identifies up to 180 classes of HR documents and can be retrained for accuracy purposes or for new classes.

Once the text is classified, the actual document is filed in the right folder and the key value pairs are extracted to provide information about the document to the HR application (5).

Challenges

As with every AI project, data is at the heart of NLP. In an enterprise NLP project, the data stored as text in unstructured formats is domain specific. There is no training data set that can replace the knowledge that a SME has about the structure of the documents, or about the language that is relevant to classify documents or to derive key insights.

Despite everybody's best intentions and understanding, the project started without a good ground truth (a set of sample annotated documents that serve as training data for the classification model). This resulted in a classification system that did not produce the expected results. At that point, the project team stopped developing additional models and engaged experts from the HR department to help. It took six months of grueling manual effort to identify and annotate all the types of documents. By the end, they had identified around 180 classes of documents. That is when they realized how many more categories there were than what they'd originally built their training sets on. Minor differences in a document that the human brain can quickly identify and classify can throw off an AI system. As we discussed in the introduction, an accurately annotated set of training documents can shorten the time to value of an AI project, especially when text is involved.

Other challenges included exchanging the image preprocessing solution for a faster one and the OCR to a more accurate one. Inadequate tools introduced delays and errors early in the process and affected the accuracy of the result. In a natural language project, in which many models are integrated, each model and step can

introduce errors that compound each other and influence the result of the following steps.

Key Success Factors and Lessons Learned

Despite all the challenges, this project is currently in production; the categorization precision of documents is over 96%, and the accuracy of the extraction of key data from the documents is at 91%. The project was successful and continues to add value to the business by reducing the manual effort required to classify the documents and reducing compliance risks.

Two key factors contributed to the success: business commitment to staff the project adequately with SMEs and microservices architecture. The business leaders could have pulled the plug on the project when the system was not producing the expected results. However, once the project team identified the need to engage the SMEs, the leaders recommitted to the project and made sure SMEs were available to work hand in hand with the data scientists.

The issues with the image preprocessing module and the OCR led the data science team to decouple the overall architecture into microservice components. This ensured that new techniques, algorithms, and products could be added or replaced as the project continued to mature and as new techniques and products were developed, without causing major disruptions to the system.

Use Case 2: Enabling Virtual Agents to Answer Questions from Health Care Providers

Health care providers regularly contact medical insurance companies with inquiries about their members' health plan benefits and eligibility, often during a patient's visit. More than 60% of the calls are related to routine, specific questions regarding the service the patient is about to receive. To reduce call center costs, the business was using an interactive voice response (IVR) system to automate calls. However, callers were so frustrated with the laborious automated menu that they would skip the automated system to talk to a real live human, as most of us do when pressed for time. The problem with callers getting frustrated is not only a satisfaction issue: every time a caller opts out of the IVR system, they are routed to an

outsourced call center, and the company has to pay the center for each call.

Business Need: Faster, Friendlier, and More Consistent Service to Providers

The nationwide health insurance company in this use case receives over one million provider calls every month. If the call center's agents are not well trained, they may not provide accurate responses or treat the caller with the friendly and respectful attitude the organization would like to offer health care providers. The department responsible for the interaction with health care providers had to find a better way to automate these calls to reduce costs.

Solving the Problem: Systems Architecture, Technologies, and Techniques Applied

This health insurer was one of the first enterprises to implement a virtual agent to respond to the questions that make up most of the inquiries (and that can be resolved with well-defined answers). You may be tempted to stop reading now. But rest assured, this isn't another story about your typical chatbot. The solution we're discussing here is really a *conversational agent*. A conversational agent, as opposed to a chatbot, is able not only to respond to common questions but also to resolve unclear user input and handle the conversation when the user changes the direction of the inquiry. A conversational agent can also use input from an enterprise application to provide the user with only the pieces that are relevant to the conversation at hand. Implementing a conversational agent in large organizations and in those with strict legal and security compliance requirements is much more complex than setting up a set of basic questions and answers in a library.

Figure 6 shows the architecture of the conversational agent solution.

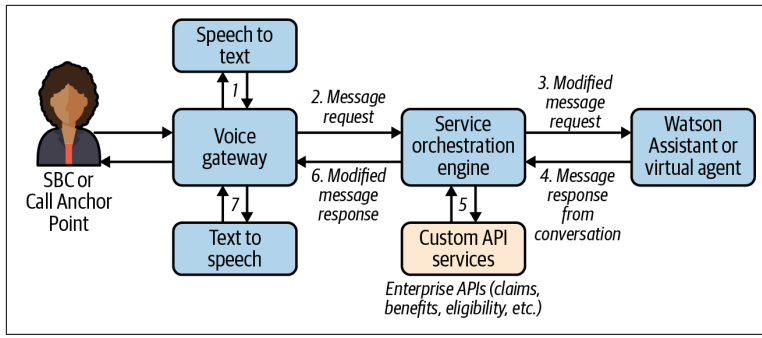

Figure 6. From voice to text to virtual agent and back to the caller

Let's begin with the voice gateway. The function of the voice gateway is to provide the interaction between the phone system and all the services to understand what the provider is calling about and return the correct next action or response.

The voice gateway first sends the audio stream to a speech-to-text service (1), which converts the caller's audio stream to text and returns it to the voice gateway. The voice gateway sends this text to an optional service orchestration engine (SOE) (2), which directs messages, if necessary, to API services to modify them (5). A modification may be required, for example, to anonymize requests by removing information such as personal health information (PHI), personally identifiable information (PII), and personal credit information (PCI) when sending the request to the virtual agent and then inserting it back into the response.

The resulting text is then sent to the virtual agent (3), which analyzes the text, maps it to intents or capabilities, and provides a response according to a dialog map. In this case, where Watson Assistant (WA) is the virtual agent, it handles disambiguation when user input is not clear, handles digressions when the user changes topics in the middle of a conversation, and may return answers from a knowledge base (4) or make an API call to a legacy system through the SOE (5). If necessary, WA will transfer the call to a live agent.

The SOE sends the modified response to the voice gateway (6), which sends it to the text-to-speech service. The service then returns an audio stream (7) with the appropriate dialog element to serve the person on the phone.

Challenges

When the project team began to design their solution, chatbots or conversational agents were not as ubiquitous or advanced as they are today, and many NLP tasks that are easily implemented today were not available then. The speech-to-text service used in their solution, for example, was built to accommodate Standard English language. When the solution was first deployed in the United States, the accuracy of the results was not high enough to meet the business objective of call containment. The reality is that there are many accents and pronunciations in the English language spoken in the US. To capture speech with the highest accuracy, the data science team annotated 25 hours of audio recordings and developed eight custom models and two acoustics-based models.

The other element affecting accuracy in the early stages of the project was the assumption that callers would provide detailed statements, including the information they would be requested to enter. As an example, they imagined a caller would state something like "I am calling to understand the status of claim number ###### for patient ID ##########." The dialog service was scripted with this principle in mind. The script, however, was complex and required the solution to be able to understand the intent, along with associated information, all in one fell swoop. This meant understanding words and numbers and combinations of numbers and letters all with one speech-to-text model. It is complex enough to understand dialog; understanding a combination of numbers *and* letters within dialog is even more complex. So the system performed poorly. Using one out-of-the-box model to pull out three or four pieces of information with different formatting did not perform well.

The team responsible for training the system, which included SMEs from the call centers, decided to take a deep look at caller dialog recordings, and what they found was that users, with few exceptions, were not making complex statements. Given the way they were used to interacting with the IVR, callers were trained to say one word, such as "benefits," or a short statement like "claim status." With this information, the dialog service was rescripted to address the real way users behaved with the system. Further, the speech-to-text system was optimized to transcribe one data element at a time. The result was higher accuracy for the users.

Key Success Factors and Lessons Learned

A key success factor for this use case was that the executive sponsor of the project was committed to its success. As expressed by the project's business sponsor, "This is not the type of technology that you set up and walk away from. It needs monitoring, training, and supervising to harness the true insights that can be brought forth." The sponsor engaged with the team, set the business direction, and followed through at every point of the iterative process. This sponsor understood that this is not a one-and-done project. It requires iterations, and it is improved at every turn.

The project sponsor's focus on business outcomes, and their understanding of what it would take to get to those outcomes, drove the success of the project. This vision provided a common objective for the SMEs, the data science team, and IT.

Two important lessons were learned in this use case:

Custom models are a differentiator for providing business value
: Pretrained models are great for general-purpose applications. However, building custom models that learn the language of the business is where real differentiation happens. There are two areas where custom models were trained in the speech-to-text service: the acoustics models that were added to understand different accents and the speech models. The custom speech models converted text and combinations of letters and numbers. The result was better accuracy in understanding member ID numbers, claim ID numbers, and medical ID numbers that are specific to the way this insurer manages its business. This required the customer service representatives to annotate or label the audio for accuracy.

Design from actual user behavior
: It is typical for teams designing new solutions to make assumptions about user behavior. However, taking the time to understand the users and their behavior can reduce the solution's time to market. In this case, the initial system was designed to account for imagined scenarios that were more complicated than required. This led to time lost on a solution that would not perform as desired. When the data science team, together with the SMEs, analyzed actual audio, they were able to design the

solution to fit user behavior and achieve the desired business outcome.

Through speech customization training, the solution achieves an average of 90%–95% accuracy on the significant data inputs. This speech accuracy helps the virtual agent handle more calls than the previous IVR system. Further, the agent also handles several subintents within the major groupings of eligibility, benefits, claims, authorizations, and referrals, enabling the insurer to quickly answer questions that were previously unanswerable. In the previous IVR system, a request for benefits could lead to a seven-page fax. Now, the conversational agent is able to respond with information about a specific benefit, such as, "The copay for chiropractic visits is $100." Because of a commitment to model customization, the project provides more detailed information, with higher accuracy and better containment, than a generic solution.

Use Case 3: Reframing the Future of Due Diligence for Mergers and Acquisitions

Merger and acquisition professionals help companies investigate the financial, taxation, commercial, operational, IT, and cyber environments associated with a company when the companies are considering a merger or acquisition. The goal is to uncover information that will guide the negotiation process and help organizations make the best possible decision.

Early on in the process, risk analysts need to make a quick assessment, looking at external information on social media, news, and financial feeds. Once the due diligence engagement begins, they need to perform a deep analysis of a company's financial, operational, IT, HR, and tax perspective by combing through a large number of documents, making sure they do not miss anything that may pose a risk to the organization considering the transaction. It is a grueling and time-consuming job. Often one number in a table or a qualifying statement can be a determining factor.

Business Need: Speeding Up the Transaction Diligence Process

In a manual approach, analysts are responsible for downloading and manually reviewing thousands of files in a nested folder structure. The ability to search for specific elements is limited, and many times, an ad hoc collaboration is undertaken within each engagement team to complete the task. The problem with this approach is that timelines for analysis are getting shorter. The amount of unstructured information to analyze is growing, and finding skilled resources has become difficult and expensive to maintain.

So the real challenge is speeding up the discovery process. But how do you speed up the discovery process when there are thousands of documents to sift through and, as new insights are revealed, additional reviews or data is required, adding more time and pressure to the process?

Solving the Problem: Systems Architecture, Technologies, and Techniques Applied

The solution is to automate the information search so analysts can focus their valuable time on the analysis and interpretation of evidence rather than on searching for the evidence. This means finding the documents that contain the information that the analyst is looking for and then extracting relevant information contained in them.

The first task was to organize the documents into different categories: finance, HR, sales, and so on. A document classifier was trained to accomplish this task.

To find the evidence, each document was then run through a machine learning model, followed by a rules-based model (we'll go a little deeper into rules-based models later in this section) to extract entities and their relations. And, finally, a Q&A system was set up with some standard queries and an index to facilitate search.

Let's take a look at how this was accomplished with the simplified architecture in Figure 7.

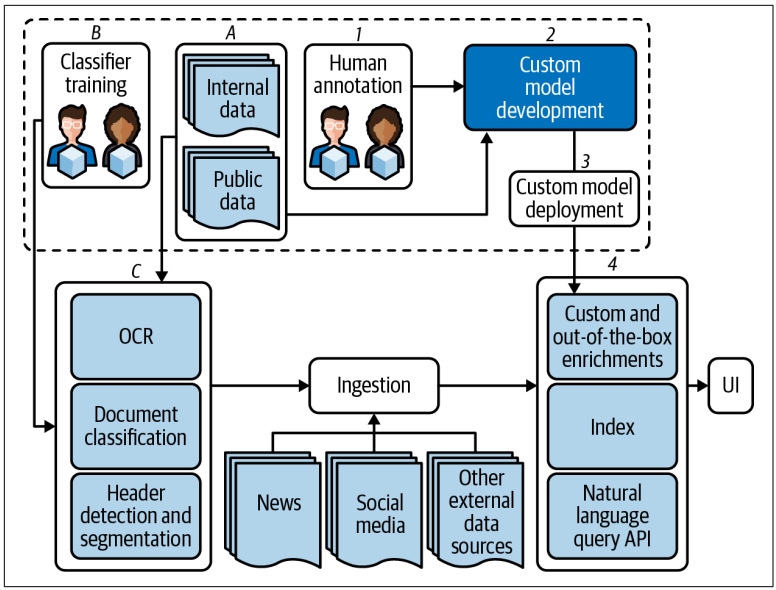

Figure 7. From public and private data to indexing data for search

We will begin with the area enclosed with dashed lines, which represents the process of training the models, and then go through the unshaded area, which represents the runtime.

First, the risk managers (SMEs) annotate documents to create the training sets for the machine learning annotator (1) and for the classifier (2). Custom annotator models are developed, in this case using IBM's Watson Knowledge Studio, and documents (A) are then annotated using machine learning models. At runtime (unshaded area), documents that are unstructured (A) are taken through preprocessing in a pipeline very similar to the HR use case (C).

The data produced in (C) is ingested, along with all other external data, in IBM's Watson Discovery (WD) (4), which encapsulates the ingestion, enrichment, classification, and indexing of all the documents. WD includes a set of pretrained models to generate enrichments such as named entities, sentiments, or keywords, and it allows for the deployment of the custom-developed models (3). The data is now ready for analysts to search, running natural language queries or preconfigured queries and getting the most relevant evidence.

Challenges

In this scenario, "teaching AI the language of your business" happened during the training of the custom models. Out-of-the-box natural understanding models such as those provided by WD and other natural language toolkits can identify common named entities and the relationships between those entities. They can also identify keywords and sentiments. But they have not been developed or trained for the specific processes the analyst follows to do their research or for the domain-specific terminology or concepts they are looking for. Therefore, although documents could be fully indexed for search, the results were not as accurate and relevant as the analysts needed them to be.

To get to the level of accuracy required to reduce the workload on the analysts, more work needed to be done. To discern whether a merger or acquisition was sound, the analysts needed to better understand the entities and relationships between them.

The data scientists sat with the analysts to learn the mental process they follow to unearth the evidence needed to arrive at the right conclusions. The data scientists needed to understand the type of questions analysts would be asking for each business area. They also needed to understand how the data they were looking for was structured.

The data scientists ultimately identified three types of questions that analysts might ask of the data:

- In the first type, certain patterns were simple and straightforward. For example, evidence relating to the concept of cost can be expressed with terms describing various types of fees and expenses. For these types of cases, they used a simple rules-based approach for which they configured a model based on a dictionary. Once the dictionary was defined, experts validated the enrichments to see whether terms should be made more specific. Because a rule-based model matches text exactly, experts only needed to review/validate a few results.
- The second type included more complex concepts that involved multiple words that might appear in various ways within a sentence. For these cases, which occurred frequently enough, an ML approach was applied. An example might be "restructuring events," which could be signaled by many statements, such as a

"divestiture," an internal reorganization, closure of a branch office, and so on. To train the custom models, the SMEs first annotates sample sentences, identifying words or phrases and the relationships between the words, highlighting the relevant enrichment that the model should identify. This created the so-called ground truth. The data science team used IBM Watson Knowledge Studio, the tool selected for the custom model training and testing, to train the custom model with the annotated examples. During the development, the model was trained with a subset of the annotations and tested on the rest. After multiple iterations of adding examples to poorly performing entities and relations, the model was trained on the entire ground truth, deployed, and applied to previously unseen examples. Then it was the SMEs' turn again to review and validate the results. Because a trained model generalizes from the trained examples, there may be surprises in the enrichment that require adding more examples to the ground truth. The trained models were tested and evaluated frequently to improve their accuracy.

- The third type included questions that, although critical for a decision, are complex and do not occur with enough frequency to be modeled. These were left out of the solution.

This was a long but critical process. This is where the expertise in the minds of analysts (SMEs) was transferred to AI systems and where much of the success of the project was realized.

With these questions, they identified which entities could be modeled using pretrained models and dictionary and rule-based models and which needed to be trained with examples. The resulting type systems required multiple iterations to hone the right annotations for the top 100 questions.

Key Success Factors and Lessons Learned

Having the right SMEs dedicated to the project was a key success factor. Only they could accurately transfer their business knowledge and expertise to the AI system, with guidance from the data science project team. This was a collaborative process. The data science and analyst teams were required to work closely for many hours to (a) understand the right entities and relations that needed to be extracted from the documents to quickly find the right evidence related to

the right questions and (b) annotate the documents for NLP model development and training.

Annotation guidelines need to be updated regularly
: To increase the accuracy and quality of the results, the team learned that continuously updating annotation guidelines with new ambiguous examples, and the agreed-upon annotations, resulted in more consistent annotations from the SMEs.

The business outcome should always be the North Star
: Whether the expected outcome is an increase in the speed of well-executed analysis or user satisfaction, the business outcome is ultimately the goal that, when met, will make a difference to the business and drive everyone's behavior and commitment to the project.

For this project, the key success measure was the Net Promoter Score (NPS)[1] of the end users for the solution. A desirable NPS ensured that what was being delivered was producing the expected business result. Of course, precision and recall (accuracy) metrics were used, but those were used to measure progress during the build and run phases of the NLP pipelines, not as the overall success metric of the project. Achieving a certain level of accuracy by itself doesn't necessarily translate into achieving specific business outcomes, and it can increase over time with the application of the learning from each iteration to the next one. So, the important thing to look for is how accuracy is *improving* over time versus a specific accuracy number.

The platform for this use case is currently in production, with analysts in all geographies conducting due diligence analysis. The system provides analysts with ready insights from the documents that allow them to more quickly find the evidence that is key to the task at hand.

1 The Net Promoter Score is an index ranging from -100 to 100 that measures the willingness of customers to recommend a company's products or services to others. It is used as a proxy for gauging the customer's overall satisfaction with a company's product or service and the customer's loyalty to the brand. For this definition and how NPS is calculated, visit Medallia.

Use Case 4: Reducing the Legal and Financial Risks Associated with Service Contracts

In our last use case, we look at a company that delivers on hundreds of service contracts in any given year. Some engagements are very successful, resulting in satisfied customers as well as positive financial results. Others, although they meet customer satisfaction measurements, result in lower than expected gross profit.

Business Need: Reduce the Legal and Financial Risks of Service Contracts

The company's Finance and Operations department initiated a project to reduce the number of engagements resulting in lower than planned gross profit (GP). The department engaged in an Enterprise Design Thinking workshop, in which they looked at the entire life cycle of a contract with a diverse team of professionals. What they learned was that despite the focus on developing standard terms and conditions (T&C) templates, a significant number of the issues came from the use of nonstandard terms in contracts, which were either not identified during the contracting process or not appropriately managed throughout the delivery of the service.

Reviewing contracts prior to signing, and monitoring them during execution, is a highly manual and resource-intensive job; it is also highly likely that something will be missed, therefore increasing the financial risk of the project. The challenge is moving contracts through the review quickly but with enough oversight to effectively manage risk. Contracts with nonstandard terms can introduce risk. If it is hard to determine if a contract contains accurate language, there is a risk of being noncompliant in addition to leaving revenue on the table.

Solving the Problem: Systems Architecture, Technologies, and Techniques Applied

The solution is to automate the review process. There are many ways to do so, but in this use case, we will focus on automating the process to identify risk factors in contracts.

The first approach used was to leverage the power of deep learning to identify common elements in contracts with unexpected revenue

losses. Unfortunately, they did not find any conclusive evidence for either successful contracts or unsuccessful ones. It seemed contracts with similar language could still have different results.

Having arrived back at square one, the next approach was to identify language that could be predictive of results different than expected. The project team engaged with the lawyers and risk managers and decided to focus on finding language that should *not* appear in a contract as well as language that *must* be included in a contract irrespective of the origin of the terms and conditions.

A schematic of the solution architecture is shown in Figure 8. If the contract is in a PDF or other image format, then as we have seen in other cases, the text is first extracted from the document so it can be sent to the classifier. Using a rules-based approach, the classifier produces a list of the contracts considered to be high risk.

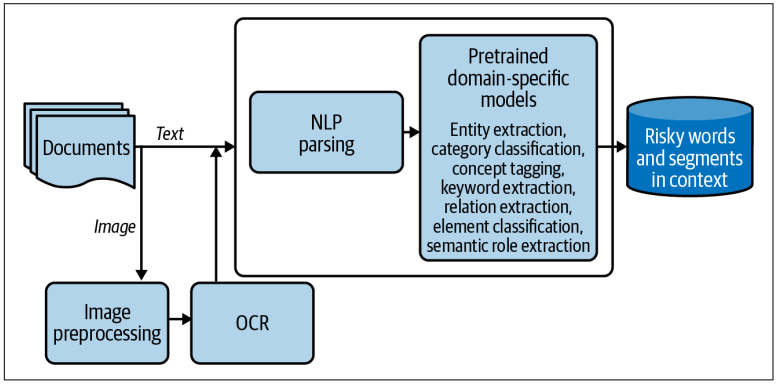

Figure 8. Identifying contracts containing words considered to increase risk in a contract

The problem with this approach was that words in and of themselves turned out to be insufficient to determine the risk in a contract, so the results were not sound. The same words used in a different context could have completely different implications.

The next step to make the system useful was to look for the words in the context of a sentence. The way the project team approached this was to identify the location of the word within the structure of a contract: Payment Terms, Billing and Invoice, Termination Clause, Statement of Work (SOW), and so on. They also needed to understand which party the sentence applied to and whether it was an obligation or a right.

They discovered a pretrained AI model that helped them categorize each contract into one or multiple contextual categories, such as Billing and Invoicing, Payment Terms, and so on. The model was also able to identify the nature of a sentence and the party to whom it would apply: whether it was an obligation, a right, a disclaimer, and so on, and whether it applied to the buyer, the supplier or the end user. In addition, the model extracted key attributes—such as currency, duration, and dates—while also providing metadata that allowed users to know what type of contract they were looking at.

Now the team could see the risky word or segment in the context of the entire sentence or clause. For example, they could see whether there was a payment term and the number of days in that term. The system was now proving more useful.

As is usually the case, this presented a new opportunity: Could this project team organize their findings for each contract so risk managers could focus on the highest-risk statements first?

To answer that question, a scoring mechanism was applied to each statement, based on the nature of the sentence and to whom it applied. This was not a trivial job; as we've mentioned, different words have different risk measures. In addition, the same word has a different risk associated with it, depending on whether it is an obligation or a right. To minimize the number of false positives, a disambiguation model was applied to the results of the rules-based scoring model. Using disambiguation helped determine the meaning or impact of the word in its particular context.

Rules-Based and ML-Based Approaches to NLP

Rules-based approaches to NLP rely on programs to list identities and relations, whether grammatical or domain specific. A machine learning approach is based on algorithms that learn to understand language without being explicitly programmed. This is possible through the use of statistical methods, where the system starts analyzing the training set (annotated corpus or ground truth) to build its own knowledge and produce its own rules and classifiers.

In other use cases for this solution, using this rules-based approach to the problem offered an additional benefit: it could be applied to many areas of the business, such as to analyze sales for revenue

recognition and accounts receivable to identify whether a contract had nonstandard late payment fees or nonstandard payment terms.

The Finance and Operations project team soon began to do risk assessment for other business units. All they needed to do was build new dictionaries. But the logic of the solutions remained the same. They expanded quickly to eight use cases within the organization. Because the solution helped them identify the obligations, reports, deliverable dates, payment agreements, resource allocations, and so on, the application also helped the services delivery team manage risk during the delivery of the project and apply risk mitigation actions.

Given the successful approach of natural language processing techniques to the problem, this team has been able to solve many other contract management needs, including making sure key terms and conditions are included in every contact and in the standard terms and conditions established for every business unit.

Challenges

As we saw earlier, when the project team first tried to identify language in the contracts that could be a predictor of success, they were unable to get any conclusive results. In this case, they did not have a big enough data set for ML to identify any useful trends. Contracts are confidential documents, sometimes with very stringent terms like nondisclosure agreements (NDAs), which limit their distribution outside of the intended audience. The team struggled to get enough contracts. This might be the case in other NLP projects that involve protected information, even when they leverage pretrained models with industry-specific information.

As in other use cases we discuss in this report, *the language of the business is defined by the subject matter experts*. Who are these subject matter experts we keep referring to? The people involved day in and day out in the business area that is trying to leverage NLP to support their customers better, to understand what customers are saying about their products, or to make their job easier. In this case, engaging the SMEs was critical to get to the right ontology for the Risk Assessment department as well as for the Accounts Receivable department. As the project continued to expand, other functions within the company wanted to leverage it. For every new department, engaging the SMEs was critical: different service lines have

different language and, therefore, manage risks differently. What is acceptable for one area may be a critical risk for another.

In this case, as in other cases, engaging the SMEs to get the keywords to build the ontology as well as to get validation of the solution's output was not easy. SMEs have their own jobs to do, and risk analysts are under pressure to review contracts quickly. Their time and dedication requirements were not accounted for in the project plan, making it difficult for them to prioritize the project team's requests, especially those more time-consuming tasks like reviewing results and providing feedback.

Key Success Factors and Lessons Learned

This was an ambitious project. The initial inclination to use deep learning to find the correlation between the language of a contract and the financial results seemed a natural one. The expectation created by the hype is that AI technologies can find hidden knowledge and trends in data. And while it is true that, fed with enough information, ML and in particular DL will find connections that humans may not quickly and easily see, they require massive amounts of data the team did not have access to. Furthermore, only the SMEs can gauge the relevancy and accuracy of the findings.

This use case is interesting and complex. The team has been working on this contract management project for two and a half years. In the beginning, the team went through some trial and error to find the approach that would work best for the problem they were trying to solve. They finally decided to use a vendor solution that included models pretrained for contract analysis. Although this approach often doesn't provide a differentiated value, it can bootstrap the solution for faster time to value.

How AI Understands the Language of Your Business

Organizations are increasingly leveraging NLP technologies to automate processes and gain insights from natural language data created and stored on the internet and in their internal systems. However, as discussed at the start of this report and as seen through the use cases discussed, the first attempt to leverage these technologies often leaves businesses disappointed with the results.

NLP technology is advancing at an unprecedented pace. New neural network machine learning models trained with vast amounts of data, such as BERT and GPT-3, are making it easier to process natural language with less training required. However, for organizations in highly regulated industries, and for many applications where explainability is important or bias mitigation is key, these might not be an option. Also, when organizations want to take advantage of the data or, more important, the expertise that res within their business, they will still have to train their own models and do so with their subject matter experts.

From the use cases reviewed, and from discussions happening in the market, we can draw some conclusions about the key elements required to deliver successful NLP projects with *faster time to value.*

Transferring Intelligence to AI

Expertise, or intelligence, is transferred to AI through the annotation process. The more accurate the annotation, the better the results of the AI system. The best, most qualified people to provide accurate annotations are the domain and subject matter experts. The annotation process is a feedback loop facilitated by the data science team.

For AI to make sense of the language of a business, it is not enough for it to understand English or Spanish or Mandarin. It is also necessary to understand *business-specific terms* and the relationships between those terms—in other words, the jargon. If we are trying to gather insights from decades of expertise hidden in the documentation of a petroleum company, we need to work with the operators at the drill site who understand what happens in the process, learn which information is relevant, and use the meaning and connection between terms so their expertise in the field can be transferred to the system. If we need to understand the impact genetics has on an illness by going through years of medical histories, we need to work with doctors, and so on.

Getting Annotation Right Reduces the Time to Value of NLP Projects

As we saw in each use case, the initial results were not as expected. The projects started to yield better and better results only after the appropriate SMEs were engaged. It follows then that SMEs should be engaged *early* in the project. This, of course, is the most expensive annotation option, and therefore careful consideration should be given to other alternatives, especially running through the first couple of iterations as well as producing the ground truth against which annotations will be measured. Some alternatives follow.

Pretrained models
: All major AI industry players and some domain-specific AI players offer pretrained models that can automate the first passes. Running sample documents through these models before handing them off to the SMEs can save them time spent annotating common terms, allowing them instead to focus on the outliers, or those terms that are very specific to your business. Consider this option carefully, because some pretrained models can introduce bias or may be unexplainable. In other cases, making changes may result in more wasted time.

Outsourcing
: With this option, you are still responsible for the result, but you can leverage a cheaper resource to do the most common annotations and leverage the SMEs only for the outliers. There are many companies that provide outsourced annotation services, and crowdsourcing is another option.

For a detailed discussion of different options for annotating text, I recommend checking out *Human-in-the-Loop Machine Learning* by Rob Munro (Manning).

The Annotation Process is Key and Must Be Collaborative

The process of transferring human expertise to the machine through the annotation process requires a collaborative approach. Let's walk through the steps of the annotation process and understand where collaboration needs to happen (see Figure 9).

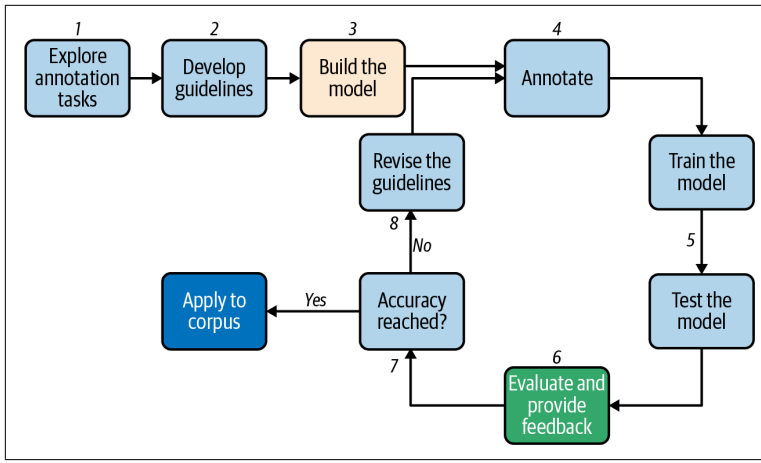

Figure 9. An overview of the annotation process

1. *Explore the annotation task.* Examine a small, representative part of the corpus[2] to get a feel for what the annotation will look like (SMEs and data scientists).
2. *Develop the annotation guidelines.* Given the ambiguity presented by language, building a consensus on meaning and categories among annotators is key. Annotators use the guidelines to decide what to annotate (which segments) and how (which category). Because there will still be disagreements, new edge cases, and so on, these guidelines have to be revised continually, as discussed in the due diligence use case (SMEs and data scientists).
3. *Build the annotation model* (Data scientists).
4. *Annotate.* Have senior members of the annotation team—the ones with the highest expertise levels in the domain at hand—annotate a small subset of the corpus to create the training set for the model (SMEs).
5. *Train and test the model.* Run another subset from the corpus through the annotation model.
6. *Evaluate and provide feedback (SMEs).*

2 A corpus is a large collection of machine-readable texts that have been produced by collecting proportional samples of representative texts/documents for the task at hand.

7. *Determine whether accuracy of the annotation has reached a predefined level (SMEs and data scientists).*
8. If so, *annotate the corpus* (SMEs and data scientists).
9. If not, *revise the guidelines*, create a new training set, and retrain the model until accuracy is achieved (SMEs and data scientists).

Additional detail on the annotation process and methodology can be found in *Natural Language Annotation for Machine Learning* by James Pustejovsky and Amber Stubbs (O'Reilly) and *Collaborative Annotation for Reliable Natural Language Processing: Technical and Sociological Aspects* by Karën Fort (Wiley).

NLP is Only as Valuable as the Results It Provides the Business

All project stakeholders need to work as a team around *common business-driven goals and objectives*.

As we saw in the use cases, when there was executive commitment to the project, the success measures for the project were business related and not AI/DS related. These measures became the common objective for every member of the project. This does more than align the team. It gives them a reason (and permission) to work together, to spend time in a room noodling through the data and problems, and to celebrate successes, which leads to the next conclusion.

Introducing Errors throughout the Pipeline

Most NLP pipelines include multiple steps and multiple models. It is important for the accuracy of the result that each step produce accurate results. Although we did not discuss in detail all the ways models were linked to produce results in our use cases, it is nonetheless an element to pay attention to. In the use cases reviewed in this document, the natural language data that businesses were trying to analyze could be stored as audio, as text in a PDF or table, as images, and sometimes even as videos. So, a critical step in the projects was the process of changing or extracting the input to text so that it could be processed.

In the HR and due diligence use cases, text first had to be extracted out of résumés and prefilled forms that were uploaded as

documents, PDFs, or scanned images. To do this, the structure of the document had to be identified first, and then the text had to be extracted. In the virtual agent case, speech recorded as an audio stream had to be converted to text. These were not trivial challenges and, in both cases, took up a good part of the effort needed to get to the expected outcome. In addition, errors introduced in this step propagated and compounded through the workflow, affecting the accuracy of the result.

Iteration

Finally, Figure 10 shows what I consider the success curve for an NLP project. Preparing your teams for multiple iterations on a solution will reduce the impact on stakeholders and boost the team's morale as well as increase the likelihood that stakeholders will invest the resources and energy required for the follow-on iterations.

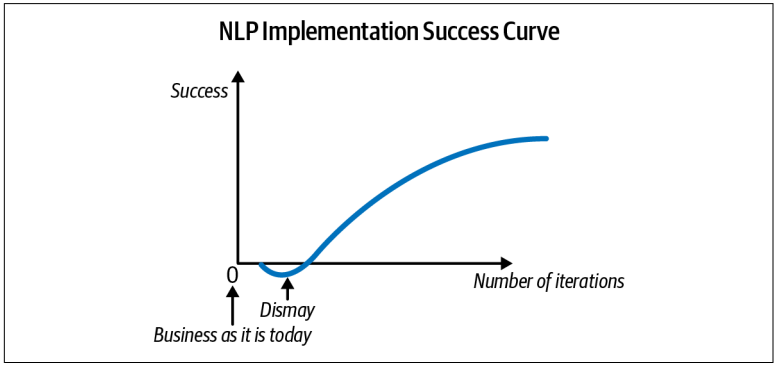

Figure 10. Typical success curve for an NLP project

Natural language processing technologies will continue to evolve, require less effort to implement, and be less dependent on SMEs. However, given the nature of human intelligence and the complexity of our social structures, the biases we have transferred to the machines and the need to explain the results of an AI system, I believe the natural evolution of natural language processing will be in the direction of integration and collaboration between humans and machines.

Acknowledgments

This report is the result of hard work by many IBM data scientists, project leaders, and architects who delivered on their projects and were willing to share their experiences with me. Thanks to Sara Elo Dean and team, Andrew Freed and team, Ales Prochazka and team, Gianluca Antonini, Anthony Stevens, Ulrike Zeilberger, and Vasanthi M. Gopal. Special thanks goes to Paco Nathan for sharing his wisdom both as a data scientist focused on natural language processing and as a writer. I also would like to thank everyone who spent their time with me and informed my thinking, even if we did not include them in this report.

About the Author

Kinga Parrott is a senior AI Strategist and evangelist who works to help people and organizations develop their data science and AI expertise. She writes blogs, whitepapers, and other materials for experienced and aspiring AI specialists and data scientists in enterprises. She is a strategic thinker who finds connections among technology, business, and people to help overcome technology adoption barriers in the enterprise.