

NLP FOR CONVERSATION DATA

HOW TO BUILD, VALIDATE & APPLY
HIGH PERFORMING MACHINE LEARNING
TEXT-BASED LANGUAGE MODELS TO
ELEVATE SOCIAL & VOC ANALYSIS

CONTENTS

INTRODUCTION

What is artificial intelligence?
What is social intelligence?
What is a machine learning model?
What can social intelligence models tell us?
How can you use social intelligence models?
Why Measurement Matters
Align Your Measurement to Your Needs
The challenge of building effective machine learning models

DEVELOPING ACCURATE SOCIAL INTELLIGENCE MODELS

Define use case
Verify sufficient data exists
Define coding guidelines
Collect conversations
Code conversations
Modify model
Assess model
Analyze errors
Step 1: Conduct an ad hoc review of the results
Step 2: Build a confusion matrix
Step 3: Prioritize the types of errors by importance
Step 4: Look for root causes and suggest changes
Deliver model

EVALUATING SOCIAL INTELLIGENCE MODELS

Understanding model evaluation
Choosing your evaluation metrics
Precision and Recall
K
Area Under the (ROC) Curve (AUC)
Using training sets and testing sets
Assessing the model against business need

SUMMARY

ABOUT CONVERSUS.AI

GLOSSARY

ABOUT CONVERSEON




INTRODUCTION

Organizations are awash in untapped, insight-rich unstructured data. Customer feedback and unprompted opinion through social media, product reviews, long-form survey verbatims, call center transcripts and more are veritable goldmines of insight for those customer-obsessed companies that can effectively harness, filter, process, and understand this massive, messy data set. Computer World magazine forecasts that unstructured information might account for more than 70%–80% of all data in organizations.

Yet organizations today face a conundrum: even as this data set grows exponentially, most brands are processing and using only a small portion of it—Forrester Research says most organizations are processing less than 21% of this unstructured data. And with some good reason: this unprompted “language data” is complex. Implicit meaning, sarcasm, slang context and much more make it challenging to separate the signals from the noise and make the data actionable in a time span needed for competitive advantage.

Today, however, a growing number of organizations are leveraging advanced natural language processing and text analytics solutions powered by artificial intelligence that are proving to be game-changers and allowing these firms to begin to fully leverage the long untapped value of this data set. But doing so requires a thoughtful and clear methodology and approach that builds on the latest data science, machine learning validation and processes. While this guide focuses largely on social media, the approaches and lessons can also be applied, with some modification, to other unstructured data sources.



Language is Data.
Social and Voice
of Customer (VoC)
data is vastly
rich in insights
and competitive

This explosion of unstructured data has provided the impetus for creating this white paper. In it, we will:

Provide readers with a clear understanding of the evolution of the technologies and how today's most advanced solutions can be harnessed effectively

Help organizations cut through jargon and marketing claims, empowering potential buyers of these solutions to separate fact from fiction and make informed vendor choices.

Outline a methodical approach on measurement that will promote transparency and adherence to emerging ethical AI guidelines and perhaps help generate discussion on the development of much needed standards

Demonstrate, as a result of the above, a clear path to unlock the full potential of this unstructured data—primarily social listening data in this case—to be harnessed and used with confidence across critical business areas, including customer experience, brand health measurement, customer satisfaction, customer care and advanced analytics.

The processes we outline can be accomplished by almost anyone with enough time, resources, open-source tools and data science expertise. However, in a world where there is a high demand for fast, efficient and cost-effective technology deployment and use, many of these processes and capabilities are being incorporated into NLP platforms (such as Converseon's Conversus.AI™ platform). These platforms allow even non-data-scientists, "citizen" analysts to implement the guidelines and approaches suggested here. The appendix of this white paper provides more information about the Conversus.AI™ platform for those interested.

We thank you for taking the time to read this and hope you not only benefit from it but also join the discussion with your feedback and thoughts.

So, let's begin.

WHAT IS ARTIFICIAL INTELLIGENCE?

The standard definition of artificial intelligence is the ability of a computer to perform a task with the intelligence normally expected of a human being. But that raises the question of just what intelligence is. And, given the difficulty of pinning down intelligence, it stands to reason that what we consider AI has changed over the years. It wasn't too long ago that the ability of a computer to suggest correct word spellings was considered AI, but now that seems so trite as to be almost a parlor trick. But it is the very ability of a machine to "understand" human language that is the key AI technology required to unlock the insights within unstructured text data, especially social media.

Not only have our collective expectations of AI grown as computing power has grown, but the techniques we use to achieve machine understanding of language have swung like a pendulum between human-centric and machine-centric approaches over the years:

Starting with humans. In the 1970s and early 1980s, expert systems created by linguists using thousands of hand-crafted rules parsed and processed multiple written languages to discern meaning.

Swinging to machines. By the late 1980s, faster processors made data-driven approaches possible for verbal speech recognition, clustering thousands of speech patterns to translate sounds into words.

Swinging back to the middle. In the mid- to late-1990s, these human and data-driven approaches came together, spurring a huge increase in the effectiveness of a particular type of AI, Natural Language Processing (NLP).

Swinging extremely toward machines. In this century, machine learning has come to the fore, building on the explosion of Big Data and culminating with deep learning approaches that can correctly recognize more patterns than ever before.

Swinging back to the middle. In recent years, so-called human-in-the-loop approaches, such as active learning, have taken accuracy to a new level.

These pendulum swings in technical approaches are reflective of the fact that human language is very complex. Yet now we are able to recognize the ascendancy of practical natural language processing technology that can solve a variety of real business problems in production settings.





WHAT IS SOCIAL INTELLIGENCE?

You've probably heard the term social intelligence, but there is a big difference between a hype-laden phrase and real business value. With the social media explosion now well over a decade old, it's time for marketers to finally extract the value out of social media conversations—that's social intelligence.

With social intelligence technology, you can identify the conversations that matter to you and classify them into standard or custom categories that allow you to see trends and make better business decisions. Intelligence refers not only to the accuracy of the model but also how well the model meets business requirements.

The benefits of social intelligence are numerous:

Reduced operational costs. A large manufacturer reduced time and resources by 90% through clean data streams that don't require human filtering and corrections.

Greater organizational adoption. With clear measurement and validation scores, social data is being adopted and mainstreamed in a broader range of areas including market research (especially brand tracking, customer satisfaction and trend discovery), customer experience (CX) and even lead generation and sales support.

Use of predictive analytics. High-precision, research-grade social data has proven to have strong predictive characteristics and can be used in advanced analytics. Instead of simply reactive and descriptive use, with measurement and effective performance, the data is proving to be predictive and prescriptive.

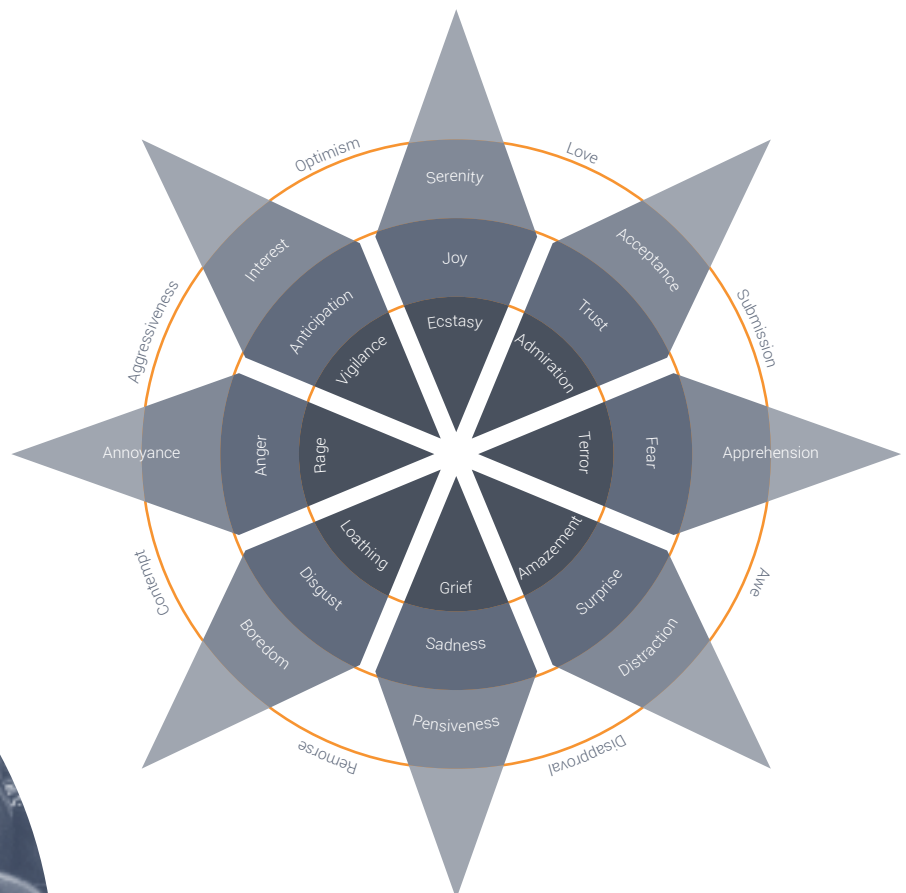
Risk mitigation. Poor or unknown data quality creates risks of making poor decisions based on data. This raises the risk of not only making erroneous executive decisions, but also can result in unintended bias in machine learning models which increasingly drive decisions in organizations.

Next, we examine how social intelligence is enabled by AI.

WHAT IS A MACHINE LEARNING MODEL?

Machine learning is an AI technique that uses a statistical model of observed patterns in input data (training data) to make predictions on new data. Generally, the more high-quality training data you feed into your model, the more accurately it can make predictions. This training data represent examples of what is already known about the situation to be predicted. For example, your model can predict social sentiment if it has been trained in social media conversations that have been labeled with the correct sentiment by human beings.

When a machine learning model is presented with a new situation to predict (such as a new social conversation to predict sentiment) it does not respond simply with the predicted answer (positive, negative, or neutral). It also provides a confidence score, which is just what it sounds like—the higher the score, the more certain the model is of its prediction. Confidence scores are important because they allow models to be tuned for higher accuracy. You might decide that you want your sentiment model to make as few mistakes as possible, so you can set a threshold for your model at 90% confidence, with predictions above 90% being used by the system and the rest is ignored. That way, only the most certain predictions are made. But that means that you might not predict sentiment for many social conversations, so setting your confidence threshold lower would provide more predictions with (likely) slightly more mistaken predictions.



WHAT CAN SOCIAL INTELLIGENCE MODELS TELL US?

Social intelligence is all about annotations—data that is added to the original social conversations to help us categorize and aggregate them. Some standard annotations that social intelligence models produce are:

Sentiment. Is the conversation positive toward your brand, negative, or neutral? Sentiment models are designed to answer that question correctly.

Emotion. Certainly, anger and sadness both convey a negative sentiment, but they might warrant different responses to the customers experiencing them. Plutchik's Wheel of Emotions, depicted here, is one common way of categorizing emotions, while other social scientists have defined alternative models.

Intensity. The strength of the passion behind an opinion can itself be measured. "It had a slight buttery off-note" is not as strong as "That is the worst-tasting \$^#% I have ever tasted."

Trust. Brand trust reflects a customer's expectation that a product or service (and sometimes corporate behavior) reflect the promises of the brand. Trust is a key quality of any relationship where customers make a purchase, yet brand trust sometimes fluctuates significantly over time. Identifying comments that exhibit brand trust can be challenging, as in the example, "I would be reluctant to use a different brand of shampoo on my infant's hair."

Innovation. In high tech, consumer electronics, and many other industries, a brand's reputation for innovation is a critical part of the buyer's decision-making. Finding the right conversations that reflect innovation is also not easy, because people often comment on things that are new, but not all of them are innovative.

Values. Brands today are expected to have a social purpose to benefit society more broadly. Consumers, especially millennials, are requiring brands to take stands on important "lightning rod" issues. By applying machine learning models to the social conversation, brands can better understand the risks, costs and benefits of engaging in the values discussions and help improve the perceptions of their CSR efforts.

Often, however, what really matters are specific insights about your unique business questions, which requires a custom model that captures specific concepts unique to your company. With custom models that classify language “like humans do”, you’ll be able to realize the full value of social data.

Buyer Journey Stages. Each company (and sometimes each product within a company) overlays a buyer journey onto its customer interactions. One company might have a four-step journey and another might have a six-step journey, and each one uses different names for its steps.

Brand Health/Attributes. Do conversations about your brand align with attributes or not, and are those attributes even relevant or in demand in social conversations?

Customer Intent. Is the customer trying to buy something or looking for customer service? Or something else? Different businesses attract people for different purposes at different times.

Emerging Trends. Social data is rich in insight into emerging trends and discoveries before they hit mainstream recognition. Machine learning models can be critical to accelerating and improving innovation at brands by finding these new trends first. And with more meaningful analysis, such as emotional analysis and econometric modeling, can help determine which ones have strong market potential...and which ones do not.

Each of these custom annotations is about the relevance of the conversations to the person who cares about them and needs to see them. Those annotations tend to be very specific to the companies that require them, while the annotations in the first list (sentiment, emotion, and intensity) tend to be more standard across industries or even all businesses.



HOW CAN YOU USE SOCIAL INTELLIGENCE MODELS?

Social intelligence can serve a number of use cases within the enterprise, including:

Advocacy. Who is engaging in conversations that “sell” your brand? As with CX, it is sometimes valuable to understand the overall numbers but also sometimes important to identify individual influencers.

Brand Tracking. Trust, innovation, and safety are all common attributes that many companies need to track on an ongoing basis.

Crisis Management. Communications professionals need to monitor breaking stories that negatively affect brand image.

Customer Experience (CX). Customer experience is the product of an interaction between an organization and a customer over the duration of their relationship. It can be measured in aggregate across many customers’ testimonials on social media or it can be segmented down to the individual level to support focused retention efforts.

Customer Service. Support teams identify complaints in social media and reach out to resolve them.

Market Research. Marketers use social media conversations to understand the wants and needs of their market.

Product Development. Product managers mine social media to determine popular product features and identify needed features.

Recruiting. Human Resources personnel can identify potential employees through social media conversations.

Reputation Management. Marketers and communications professionals assess the brand image across the social media population.

Sales Leads. Salespeople use social media to identify potential purchasers for their offerings.

Knowing your use cases is critically important, especially when it comes to the accuracy needed in your social intelligence models. Use cases that require the aggregation of data need far higher accuracy than those that depend on using individual social postings, as shown below.

WHY MEASUREMENT MATTERS

It's often said that you can't manage what you can't measure. For many years, the quality of the data processed through social listening platforms has been opaque at best and disappointing-to-unusable for insights at worst. It has been difficult, if not impossible, for analysts to clearly measure the accuracy of data in their social listening platforms. As a result, the adoption of social intelligence has too often been stunted by a lack of trust in this massive, messy, unstructured dataset. Market research professionals often look at social data with skepticism because of concerns about accuracy. Senior executives naturally hesitate to accept insights and findings without a clear understanding of the true, quantitative nature of the data.

And with good reason. "Accuracy"—how well systems match the consensus of humans—has often been only slightly better than a coin flip. Additionally, many technologies miss many customer opinions, leading to well-deserved hesitation by market research professionals who cannot effectively integrate this data into advanced analytics models, or use the data to report on key trends to senior executives.

There is good news, however. By directly learning from humans, machine learning algorithms are beginning to unlock the full value of this massive, real-time insight resource.

Accuracy measurement is the key to utilizing this data and technology effectively. Measurement tells you how accurate the predictions coming from your model are, but more importantly, it provides data-driven evaluation to understand if your models are generating tangible business value, helps instill the confidence your stakeholders need to mainstream this data, and helps ensure you are adhering to ethical AI best practices.





ALIGN YOUR MEASUREMENT TO YOUR NEEDS

Not all social listening analysis requires the same level of accuracy. Understanding when high accuracy is needed—and not needed—allows you to make investments in the right areas and not over-engineer a measurement solution when it's not needed. Conversely, this understanding prevents you from under-investing in building a robust model only to find later that it fails in damaging ways.

Often, accuracy can be “good enough” for your purposes. Quick, “quick and dirty” insight is useful in day-to-day social listening analysis when directional findings are sufficient. General reactions to a news announcement or one-off dives into the newest reactions to a product launch or advertising campaign can each employ machine learning analysis without strict accuracy measurement. Sometimes you can take a qualitative rather than a quantitative approach.

Let's take an example of a model that is designed to identify customer complaints that should be reported to customer support—this use case focuses on individual social media posts. If the model misses a few social conversations that it should find, maybe that's not that bad. Sure, it would be better if it found all of them, but if it finds the vast majority, that would be good enough for most companies. And if the model mistakenly sends a few posts to support that are not actual customer complaints, the support person assigned the case will very quickly look at it and close it, only wasting a few seconds. Even if 20% or more of the cases are erroneously flagged, most companies could live with that because a human being can quickly scan the erroneously flagged cases and remove them from the support queue. Most importantly, this approach is still likely to deliver far more complete (fewer misses) and far more accurate (fewer erroneously flagged cases) results than a keyword-driven or rule-based approach.

But what if you are doing market research? Market research requires accurate aggregated data that adds up the sum of the individual postings. If you are trying to track the sentiment of your brand mentions over time, it's quite difficult to live with 20% of the data being wrong. Suppose you miss 20% of the mentions you want and you 20% of your dataset consists of erroneously included “false positives”. Then, suppose your sentiment analysis is only 80% correct. This compounding of errors makes it quite difficult to trust your final reporting, or compare data over time to draw any conclusions. Unlike the customer support use case described above, no human being is looking at the data to weed out the errors. You now find yourself in the position of the unlucky market research professional described earlier, stuck with inaccurate data and unable to integrate it into advanced analytics models or use it to report on key trends to senior executives.



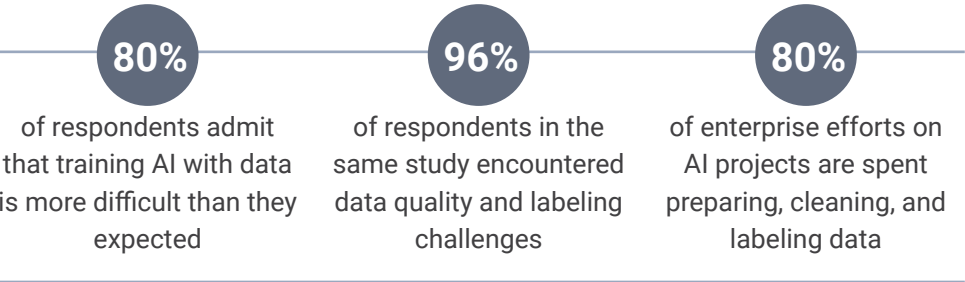
So, while you prefer your data to be highly accurate for every use case, for some use cases it is absolutely essential, while for other use cases you might be able to deliver value with less accurate social intelligence data. One way to think about this is to ask the question, “what is the impact of being wrong?” For crisis management, if the system misses even 30% of the social media posts about that crisis story, you’ll still get the idea that there is a crisis and you’ll respond appropriately. In this case, it is a business decision about how much you want to pay to be more accurate when cheaper, less accurate, tools might be “good enough.” But what if you are designing a new product and desperately need market feedback on which features are critical to success? Millions of dollars might ride on such decisions. How much less do you want to pay for the social intelligence tool that gives you the wrong answer?

The lesson in all of this is simple: take great care when choosing a social intelligence solution when high accuracy is required. Later in this paper, we’ll show you how to evaluate models so that you can measure model accuracy effectively and make the right decision. But first, we need to review the process of developing social intelligence models.

QUALITATIVE ANALYSIS “GOOD ENOUGH” MODELS	QUANTITATIVE ANALYSIS HIGH PERFORMANCE MODELS
Quick peeks + directional insight	Quantitative analysis
Usually one-off analysis	Ongoing-longitudinal scoring/analysis
Analyst-driven	Subject matter designed according to consistent expert frameworks
Statistical and proportional focus can usually be sufficient	High per document accuracy including at “target level” for root cause analysis
Brand-specific, limited application of model	Industry or competitive benchmarking, broad application of model
TYPICAL USES	TYPICAL USES
News event reaction	Brand tracking
Product launch feedback	Customer experience analysis
Crisis Management	Predictive + advanced analytics
Competitor Analysis	Customer sat scoring
	Trend discovery

THE CHALLENGE OF BUILDING EFFECTIVE MACHINE LEARNING MODELS

Machine learning for model development is a broad discipline that encompasses many different techniques and approaches. Despite the promise of machine learning for solving seemingly intractable social media insight problems, getting there isn't easy. Consider these sobering findings:

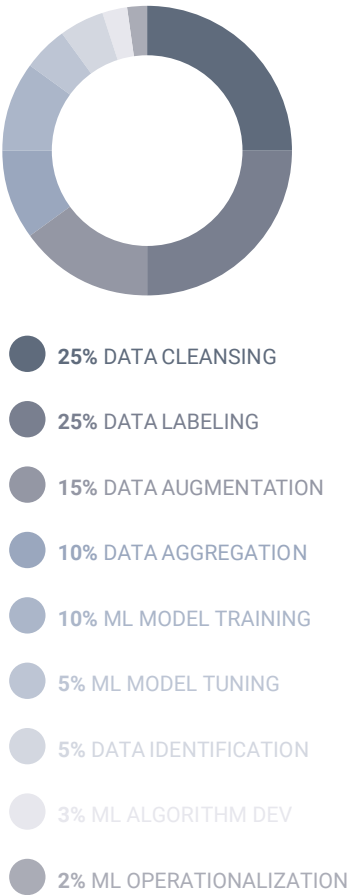


A study states, "The many steps involved in data collection, aggregation, filtering, cleaning, deduping, enhancing, selecting and labeling data far outnumber the steps on the data science, model building, and deployment sides."

The main reason that these challenges arise is that machine learning models are still relatively new, and the knowledge of how to manage the data to properly and efficiently build them is not widespread. In the remainder of this paper, we share what we know so that you won't be surprised at the difficulty—and you will have the knowledge to take on the challenge.

Models are only as good as the data they learn from and feeding the ML algorithm the "right" data is not a simple task. Analysts often suffer from a wide range of biases, from confirmation bias to cultural biases. When it comes to language, context is critical and "labelers" are going to view conversations through the prism of their own experiences and cultural norms. If the labeler has inside knowledge about an organization's business and strategic priorities, this can be a great advantage. But it can also predispose them to confirmation bias, mislabeling their training data in a manner that they know aligns with their executives' preconceived notions, preferences and general views of the world.

PERCENTAGE OF TIME ALLOCATED TO MACHINE LEARNING PROJECT TASKS



On the other hand, working with third-party data labeling services can lead to enormous quality problems, as data labelers often lack the knowledge or the incentive to label carefully and accurately, and methods for ensuring the quality of their labeling are often insufficient despite best efforts to the contrary.

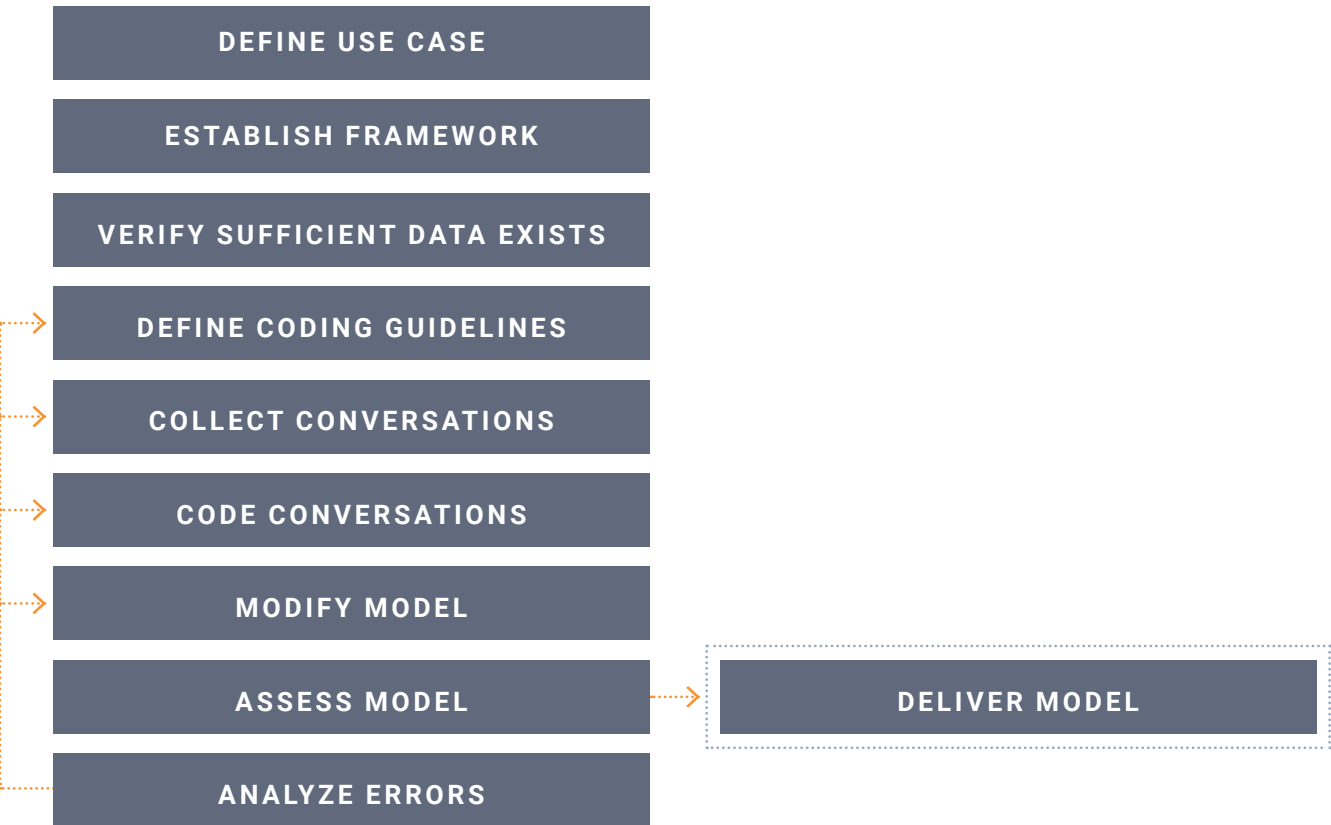
Coding conversations to ensure accuracy is a process that is still evolving but is critical to the overall impact and value of social listening initiatives. This will be covered in greater detail later in this paper.

integrate it into advanced analytics models or use it to report on key trends to senior executives.



DEVELOPING ACCURATE SOCIAL INTELLIGENCE MODELS

There are several ways to approach machine learning, with the most common being supervised machine learning—where training data is used to create a model that can predict the right answer with data it has not seen. A sample process is shown below, and a few of the steps are very important to the evaluation process, which we discuss in detail later in the paper. You’ll notice that the steps are often taken more than once—when the model is evaluated and errors are identified, you can return to an earlier step in the process to make a change that is intended to improve the performance of the model when it is tested again.





DEFINE USE CASE

The very first step in the process is to make absolutely clear which business problem (or use case) you are tackling. In so doing, you must identify the data to be analyzed by the model, the outcome that the model must predict, and the means to determine the model's effectiveness. For example, the model might attempt to identify sentiment toward your brand as expressed on Twitter each month, so the data would be tweets, the outcome would be sentiment, and you might want a .80 F-measure to be achieved in testing before using the model for making business decisions. (Later in this paper, we'll discuss F-measure and other means of evaluating models.) It is also at this juncture that you can determine the quality of your model—whether it is for general, directional insight (qualitative analysis) or requires more robust performance and validation (quantitative analysis).

VERIFY SUFFICIENT DATA EXISTS

Before going any further, a critical step to take is to ensure that the data available to the model is adequate for the use case at hand. For example, if there are only a few dozen tweets about your brand each month, then perhaps a model identifying sentiment in Twitter is doomed to failure before you even begin. You might need to expand your horizons to collect data from more sources beyond Twitter, or decide to report every quarter rather than monthly, or look for mentions of all brands in your industry, not just your own. In most situations, you will find you have sufficient data to proceed with your original desired use case, but taking some time to validate data volume can save a lot of wasted effort.

DEFINE FRAMEWORK AND CODING GUIDELINES

The most important step in acquiring accurate training data for your machine learning model is creating a well-crafted set of instructions for labeling (or coding) the conversations. The goal is to develop a set of instructions that removes as much ambiguity from the task as possible, so that the differences in human judgment are based solely on a difference of opinion. An ambiguous task, or one that is difficult for humans to perform, will not result in a highly accurate model. For example, in sentiment analysis, it might seem difficult to correctly classify this tweet: So happy to hear that Uber's stock price fell last week. Its sentiment is negative about Uber despite the fact that the speaker is happy. Tightening the coding guidelines so that the instructions cover this case (and many other ambiguous cases) improves coding accuracy, which results in more accurate data and reporting down the line.

Depending on what you want your model to predict (the classification task), you will need to devote varying amounts of time and resources to researching your concept definitions, incorporating ideas from pre-existing frameworks, and interviewing key stakeholders who will ultimately depend on your model for its predictions. In short: never define your coding guidelines in a vacuum.



The most important step in acquiring accurate training data is creating a well-crafted set of instructions for labeling (or coding) the conversations.

EXAMPLE

For example, the concept of "Trust" is an ambiguous term and stakeholders in your organization are likely to have strong views about the definition. "Net Promoter Score (NPS)," is less of a concept and more of a metric which leaves less room for disagreement. These are both critical to incorporate these into your coding guidelines, so that your model's predictions reflect the same understanding of these concepts/metrics that are held by the people who will ultimately be consuming this data. Taking the example a step further, a concept like "corporate social responsibility (CSR)", is not only widely understood, but also highly dynamic, with new, distinct topic areas being assigned to it every year. Where sustainable, environmentally friendly supply chains may be the top topic this year, workplace equality issues may take over the conversation next year. This means that your coding guidelines for such a classification task may require even more frequent updates and adjustments than for a task like sentiment or trust, and your training set will very likely require more frequent enhancement.

Beyond conceptual definitions, your coding guidelines should also include a large number of real-world text examples. Your coders (the people who will be labeling your training data) may often be the best people to collect these. Critically, their subject matter expertise will allow them to foresee and identify "edge cases"—text examples that are ambiguous and will be most likely to stymie coders when they're labeling training data.

In sum, there is a big difference between building a simple model and building an "intelligent model" that has been strategically designed and infused with true subject matter expertise. Brands too often develop their frameworks and definitions in a vacuum. Building a model that reflects a wide range of established, "best-practice" definitions, as well as the views of your key stakeholders, is extremely important to effective social listening.

COLLECT CONVERSATIONS

Many methods exist to collect social conversations. Some collect conversations as they happen, while some others can collect historical conversations to allow you to study changes over time. Some social listening tools, such as Brandwatch, can collect relevant conversations in real-time and can store them for use at a later date. To access historical conversations, you usually need to use a paid social API, such as Twitter's Gnip. Part of the collection activity often includes curation, where conversations are selected to eliminate noise, such as duplicate or off-topic conversations, and steps are taken to ensure that the conversation sample is representative of the larger set of conversations the model will work with.

In general, there are three types of social conversations:

- 1 Private.** Most Facebook, LinkedIn, Instagram, and Snapchat conversations are private by default and not available to social media collection methods.
- 2 Public.** Twitter, YouTube, blogs, and most message board conversations are public by default. If you don't need a password to access the data, it is probably public and can be collected.
- 3 Restricted.** Some conversations, such as ratings and reviews, are publicly available to view, but are restricted by legal terms and conditions that prohibit its collection for use in social media analysis. In many situations, you can pay for the privilege of collecting these conversations, and depending on your use case, that might be recommended.
- 4 Proprietary.** Proprietary VoC sources are collected and owned by the client. They can include call center transcripts, long form survey verbatims and other unstructured data.

Conversations must be collected and coded to create the training data to train the model, but also must be collected when the model is in production use.



“CODE” CONVERSATIONS

As mentioned above, creating the training data for social intelligence models is usually a manual, analytical task. Human coders must follow the coding guidelines created in the step above to label each conversation. For example, human coders improving a sentiment model will label each social conversation as positive, negative, or neutral.

But using a single coder to label each conversation is not enough. The highest quality training data uses a technique called inter-coder agreement. The standard way to do this is to take a reasonably-sized subset of the conversations and ask two to three human volunteers to code those conversations independently of each other. This is an important approach for several reasons:

Conversations coded the same way by multiple people are higher quality. It's more expensive to ask two or even three people (to break ties) to code each conversation, but it is extremely important to train the model with correct data. Incorrect data will also train the model, but train it for the wrong answer.

High rates of disagreement might indicate problems with the coding guidelines. If humans can't agree with each other, they might need to modify the task instructions to make it simpler. High disagreement could also indicate that the intended use case is not well understood.

Inter-coder agreement sets the bar for the model's potential. The percentage of time coders agree with each other is a good estimate of the best one could expect an automatic system to perform.

It can be tedious work, but correctly coding social conversations is one of the most important steps needed to create models that perform well. And “performing well” isn't just about accuracy. It's also about avoiding real or perceived bias.

Models sometimes unwittingly reflect bias inherent in larger society. For example, Google's Word2vec is a useful tool to broaden machine learning training data from the words in the training data to related words, which allows the system to recognize broader concepts that use related words. So far, so good. But what if the documents used to train Word2vec (Google News) are rife with sexist concepts? Word2vec relates the word doctor with male words (man, he, him) and the word nurse with female words (woman, she, her). Often, explicit steps must be taken to remove such bias, ranging from applying after-the-fact adjustments to the model to carefully selecting a diverse set of document coders—perhaps models are only as unbiased as the coding guidelines, human coders, and data that goes into building them. Recognizing these issues, several countries, including the US and China, have released updates to their national strategies on the responsible use of AI.

Even with single human coders or analysts, inadvertent bias is often prevalent. From cognitive bias to confirmation bias, there are a wide range of places where single human coders can go astray without proper oversight and process.

With great power comes great responsibility. Today, machine learning models—including the language models we discuss in this paper—are the state-of-the-art methods that have become the dominant means to process and analyze social and other voice-of-customer data, such as survey verbatims, call center transcripts and more. These models provide insights from which products are built, company policies and social initiatives are designed, and business decisions are made.

As mentioned above, we cannot manage what we cannot measure, so ensuring accuracy, transparency, and fairness is fast becoming paramount. Machine learning models are only as good as those who create and deploy them. Building effective models require a thoughtful, systematic and strategic approach to mitigate potential unintended bias and to exceed human performance consistently. And regulatory authorities and brands are taking note and taking action. Europe's GDPR regulations, while not specifically addressing social data, put a clear emphasis on accuracy and transparency. Ethical AI standards are being rapidly adopted by many leading organizations and emphasize the need for fairness and the ability to intercede and modify models rapidly if they appear to display bias. There simply is less room now for inaccuracy and opaqueness in social listening analysis. **As Google writes in its Ethical AI standards document:**



"AI algorithms and datasets can reflect, reinforce, or reduce unfair biases. We recognize that distinguishing fair from unfair biases is not always simple, and differs across cultures and societies. We will seek to avoid unjust impacts on people, particularly those related to sensitive characteristics such as race, ethnicity, gender, nationality, income, sexual orientation, ability, and political or religious belief."

By applying careful, rigorous, and systematic data practices, organizations can accelerate the use of data, reduce operational costs associated with messy data, get to better and cheaper insights quickly, and finally, leverage the full value of this massive source of insights. The stakes are high.

As we say at Converseon:

"In an era where the collective voice of customers and citizens, empowered through social channels, has become a primary agent-of-change in transforming governments, societies, industries, brands and products, there is arguably no greater obligation for our industry than to effectively, thoroughly and accurately capture, analyze, report, and act on these needs, wants, experiences, hopes, opinions...without inadvertent discrimination or bias."



After the difficult work of coding the conversations, we move on to the actual production of the model.

ESTABLISH MODEL

The next step is to create (the first time) or modify (each subsequent time) the model based on the coded conversations and any other steps that have been taken to assess the model and identify errors. Some of the ways to modify a model to try to improve it include:

Changing the threshold for confidence levels. You can change the tipping point where a conversation is considered relevant in a model. Moving the threshold up and down adjusts the tradeoff between missing correct answers and including incorrect ones.

Adding training data. Machine learning is typically improved by adding more training data that allows the model to detect more varied patterns that are associated with the human-assigned labels. As long as the training data is of high quality, the more data, the better.

Modifying the coding for existing training data. Sometimes, errors are detected that can be traced back to erroneously labeled training data. Sometimes that requires merely re-coding the data but at other times changes to the coding guidelines are required.

You can use each of these techniques and more to improve the performance of your model.

ASSESS MODEL

After making changes to the system, it's time to test—or re-test. As we describe in detail later in this paper, you separate the test data into two groups—one containing the data that you use to train the system and the other containing the data used to test the system. The reason to do this is that most of the changes you make to try to correct errors will turn out to be ineffective. Most fixes don't work and might even cause the accuracy of the system to regress by breaking things that once worked. It's best to get those bad ideas out of the system in development by using only a small part of the data. That way, you've saved most of the data to test the subset of the ideas that really seem promising. Once you have found ways to improve the system, then you return to the appropriate earlier step in the process and go through the analysis all over again.

The major focus in the rest of this paper is on quantitative evaluation, which is the best way to assess the model objectively, but it is also important that the model appears to work well to actual human beings. Assessing qualitatively ensures that the problem is being solved in a way that is “good enough” from a human's viewpoint and therefore likely to win the trust of the humans who are ultimately using the models.



ANALYZE ERRORS

Every model will contain errors, and yours is no exception—but perhaps it can include fewer errors than everyone else's. In order to do that, you need to analyze the performance of each model so that you can determine the source of the errors and take steps to address the root causes of those errors in your next version of the model.

Error analysis is best performed as a rigorous task that follows the same steps each time so that it is easier for us to compare our latest model to previous ones. But that rigor does not mean that it is rote or automatic—it is critical for the analyzer to constantly be on the lookout for insights that explain the “why” behind the “what.”

Our approach to error analysis consists of several steps:

1

Conduct an ad hoc review of the results

2

Build a confusion matrix

3

Prioritize the types of errors by importance

4

Look for root causes and suggest changes

STEP 1 | Conduct an ad hoc review of the results

The first step in error analysis is to eyeball the results to see if they pass the “sniff test.” The very first thing we should do is to make sure that our measurement system is working properly and that the aggregate results that we got seem to make sense when we scan through individual responses. It’s enticing to run the test and immediately look at the results without stopping to consider whether those results are reasonable, rather than the result of a measurement error.

Often, you can spot some kind of measurement error right away, which should cause you to redo the measurements before moving to Step 2. Omitting this step might cause you to analyze or even fix errors that either don’t exist or aren’t that important. If you don’t see any significant problems with the measurements—in other words, you believe the results that you got—then you can move on to Step 2.

It’s important not just to know how accurate the model is, but also to know the kinds of errors that seem to be occurring frequently. Adding training data that address these “blind spots” in the model can be a very fast way to improve. We recommend that you look for social intelligence technology that uses active learning in developing models to make that process easier—it requests more feedback for conversations that it is unsure about—but if you are seeing similar errors happening over and over, finding and coding more training data can quickly address that problem.



STEP 2 | Build a confusion matrix

Some kinds of errors are more egregious than others—that is one of the basic principles around tuning the system to reduce “howlers”—the kinds of errors that are so bad that they make users cry out in pain, and undermine their confidence in the whole system.

To categorize the kinds of errors you are finding, you first need to build a confusion matrix. For sentiment analysis, for example, it’s not enough to know that you are 88% accurate—what you really want to know is what kinds of errors are occurring. Has the model been confusing positives with negatives (awful) or positives with neutral (not as bad)?

To determine the percentage of errors attributable to different types of confusion, use a confusion matrix:

	PREDICTED POSITIVE	PREDICTED NEUTRAL	PREDICTED NEGATIVE
ACTUALLY POSITIVE	294	92	37
ACTUALLY NEUTRAL	97	416	109
ACTUALLY NEGATIVE	17	48	237

While confusion matrices can be quite useful—we’ve shown a simple example above—this example contains only three values. Building a confusion matrix for classifications with many values, such as emotion, causes the number of cells in your table to rise dramatically, making it harder to see the forest for the trees. In such cases, it can help to also prepare a confusion matrix that uses a hierarchy to reduce the number of cells, so that the most important types of errors stand out. For emotion, it is common to group positive and negative emotions so that anger and disgust are both called negative, for example. This approach lets you zero in quickly to see if the system is regularly confusing positive emotions for negative emotions. You can also pair emotions that are adjacent to each other on Plutchik’s wheel, because they are more similar to each other than those far apart on the wheel.

You can see from this confusion matrix example, above, that the predicted results generally match the actual classification, but that there is a bigger problem with positive items being misclassified as negative than negative items classified as positive—and that most of the errors confuse positive or negative with neutral rather than with each other. That’s our next step.

STEP 3 | Prioritize the types of errors by importance

Two things can make an error important:

- > **Egregiousness**—how bad is it when it happens?
- > **Frequency**—how often does it happen?

The most egregious errors that happen most frequently are the ones to focus on, because they will give the biggest subjective and objective improvements to the system.

Frequency is relatively easy to identify, given the confusion matrix, but egregiousness is much more of a value judgment. In general, it makes sense for you to prioritize the most frequent errors, but if you see something especially bad, it can be worth making it a high priority. For example, in your confusion matrix for emotion, you might decide that confusion between anger and disgust is not very important, even if it is frequent, but if you see a significant rate of confusion between anger and joy, you might want to concentrate on that because of how extreme the error is.

STEP 4 | Look for root causes and suggest changes

At this point, you must switch from looking at aggregated statistics to individual items. Start with the highest priority type of error that you chose in Step 3 and then start classifying each individual error by what you believe is the root cause. For example, as you examine errors in sentiment analysis, you might find several possible reasons that account for most of the errors, such as:

- > **Negation.** Example: “Just got off the phone with Verizon. Not happy.” The model sees the word “happy.” If the system believes this is positive, then the problem you have is that the system has failed to recognize negation in this case.
- > **Sarcasm.** Example: “Another great support call with Verizon over my lovely phone. Could this day get any better?” If the system predicts this as positive, it is likely that it did not recognize sarcasm.
- > **Slang or New Vocabulary.** Example: “Verizon just punked me again.” If the system sees this as positive, it’s possible that it does not recognize the word “punk” as negative.
- > **Wrong Target.** “So annoyed. I just dropped my new Verizon phone into the toilet.” If the system sees this as negative toward Verizon, it is likely because it did not correctly identify the target (or source) of the annoyance.

For each type of classification (sentiment, emotion, intensity, and others), you should come up with some standard root-cause categories that you can use to bucket the high priority errors that you see, so that we can determine what types of causes seem to cause the most high-priority errors.

DELIVER MODEL

Despite our emphasis on using numerical measures of model performance, in the end, the decision to begin using a particular model in production is a subjective one. In some situations, what seems like low accuracy might be an improvement over the current model or current human process, and be well worth promotion to production. Other times, 90% accuracy, which sounds quite good, might not be enough when a single mistake might be very risky or dangerous.

You can be excused for feeling as though this is a difficult process that requires expertise, investment, and hard work. If that's how you feel, you're getting the idea. That's why it's important to use tools, such as Conversus, that automate much of this work for you and put you, the subject matter expert, in charge. Conversus.AI includes a growing range of programmatic API integrations so that the models you build can be deployed immediately into many social listening, management and business intelligence platforms.





THE RISE OF EXPERT-DESIGNED MODELS

Today's models are only as good as their creators. Models "learn" from humans, including their knowledge, expertise and bias. This means making sure the best people – those with true subject matter expertise and not just technical skills – in your organization are developing the models is critical. This means getting the model development process as close to the center of knowledge as possible. Each degree of separation between this expertise and your model development process usually contributes directly to lower model performance. Are the individuals building the model the most knowledgeable about the category or business application? While computers can calculate data at the scale and speed far beyond human capabilities, humans surpass computers in the ability to simplify – that is, reach through all the noise to find the essence of a matter. Combining these skills is core to building highly effective models. If you are building a model for customer care, for example, leveraging the intelligence of those who are deeply immersed in your customer care issues is essential. If you want to build a model for trend discovery for hair care, for example, having a subject matter expert who is fully fluent in the hair care category should be utilized.

But how does one bring these subject matter experts, many of whom are often time constrained, into the process? MLaaS platforms, especially those that leverage "active learning" techniques, are purpose-built to support this expert-driven model development process, democratizing access to machine learning technology beyond data science teams and ensuring that the true subject matter expertise makes its way into your models in

the form of carefully labeled training data. Active learning is, as defined by Wikipedia, a special case of machine learning in which a learning algorithm is able to interactively query the user (or some other information source) to obtain the desired outputs at new data points. In statistics literature it is sometimes also called optimal experimental design. This keeps humans at the center of the model development allowing them to "teach" the model in the most efficient manner.

At Converseon, we employ a methodology that taps into leading experts during the development, deployment, and refinement of advanced and custom models. Brands' domain experts, who know the customers, product, and market best, directly access the technology to train models, measure model performance, analyze errors, and make iterative improvements to training data. This approach is giving rise to what we call "Expert Built Models," – that is models that have been designed from the bottom up from the best subject matter experts in the world utilizing the approaches and frameworks described here. These models are increasingly prebuilt and available to organizations for immediate use which reduces costs, assures high performance and accelerates their adoption in organizations.

Or as Forrester Research writes, "While many open source machine learning platforms are available, the cost to successfully implement them and produce useful models can range into the hundreds of thousands or even millions of dollars... Using prebuilt models from cloud-based platforms can be much more cost-effective."



EVALUATING SOCIAL INTELLIGENCE MODELS

Evaluating performance of social intelligence models can be challenging. Human language is complex. Single human analysts often disagree with each other. In this section of our paper, we provide specific methods to help you evaluate your models in an objective way.

UNDERSTANDING MODEL EVALUATION

The most basic way to judge a model is whether it gets the answer “right.” But what is “right”?

For sentiment analysis, for example, if the model says that a tweet is positive, is that really true? We automatically assume that we human beings can properly judge the answer to that question. In truth, for most tasks, individual human beings are not 100% accurate. In fact, in our experience, individual humans often agree only from 65% to 85% of the time.

And, you might ask, how do we judge that? How do we know that a person got it wrong? Establishing ground truth for performance testing requires having multiple people (three is recommended) evaluate a record for accuracy, with an adjudication process if there is disagreement. As we described earlier, this technique is called inter-coder agreement. The idea behind this concept is that if three people all agree on an answer, then it is highly likely to be correct. So the way that you set out to evaluate the model is against ground truth—the agreement of multiple people with the answer, rather than the subjective opinion of one person.

It is also important that this analysis is done independently so that one analyst does not influence another. We need the people to agree (or disagree) independently. If one person answers that a tweet is negative, and you then show that tweet to another person, asking if they agree or disagree, you have framed their evaluation. The second person is much more likely to agree. Instead, you must ask the second person to look at the tweet and form their own independent judgment of the sentiment without seeing the original person’s opinion, repeating that process for a third, and even more people as needed. This is a variation on the double-blind technique that has been adopted in other uses, including by Google as part of its human quality scoring.



You can imagine that if you show two people a tweet that says “Coors is the best lousy beer” that they might disagree on whether that tweet is positive or negative. Clearly, some tasks are easier to “get right” than others. This is important when evaluating a model, because when a person disagrees with another person they usually agree to disagree or chalk it up to different points of view, whereas when a person disagrees with a machine, they tend to say that the machine is simply wrong. In reality, the machine also reflects biases and opinions of its own based on what it’s been taught. If the machine is taught by the common opinion of many different people, it will likely be able to outperform any individual person in accuracy.

But getting the answer right is not always the only standard to shoot for. When the system makes errors, you might prefer that they be small ones—for example, it marks a negative tweet as neutral rather than positive. Both answers are wrong, but one is diametrically opposed to the correct answer and the other is at least “close.”

CHOOSING YOUR EVALUATION METRICS

There are many types of model performance metrics that you can use to assess how well your model is working:

PRECISION AND RECALL. These are the best understood metrics with the longest history, but not the only ones to consider. Precision and recall measure the percentage of true negatives and true positives against the correct responses and both can be combined into a single number, known as F-Measure.

K. K is a recently developed measure of classifier performance similar in some ways to the more common F-measure, in that it can sum up the evaluation of a model into one number.

AREA UNDER CURVE (AUC). AUC actually stands for “Area Under the (ROC) Curve.” A ROC (Receiver Operating Characteristic) curve is a graph of True Positive Rate vs. False Positive Rate across a range of threshold values. AUC might be best used to compare two models to each other.

FINE GRAINED LEVELS OF ANALYSIS. Here we are referring to the signals gleaned from a specific dataset for analysis. While not an industry-standard “metric” for performance, it is nevertheless quite important (and frequently overlooked) for social intelligence use cases. In short, many listening platforms and language models conduct analysis on a “document level”, assigning one prediction or annotation to an entire text document (a tweet, a news article, a forum post, etc.) Target or aspect level analysis, by contrast, captures and annotates all expressions of opinion toward any entities of interest within a given document, thereby delivering a far more granular set of predictions than document-level analysis. Converseon research shows that document-level analysis misses approximately 60% of available “signals” (discrete expressions of opinion) on average. Target-level analysis is therefore essential for root cause driver analysis and for advanced modeling of data. As a general rule of thumb, more signals/ annotations out of a data set represents more opportunity for insight and accuracy.

PRECISION AND RECALL

We've been using the simple term accuracy, but there are actually two aspects to accuracy that are critical to understand:

PRECISION. How many of the items predicted by the model for a specific label are correct answers, in the case of sentiment analysis—or how many items are relevant, in the case of other models. As depicted in the diagram at right, you can think of the correct answers as true positives, with precision calculated by dividing true positives by predicted positives.

RECALL. How many of the total actual correct or relevant answers have been predicted by the model. Recall is also known as the True Positive Rate (TPR), calculated as true positives divided by actual positives.

This is important, because every model yields a tradeoff between precision and recall. For example, if the model predicts one item and that is a correct answer, its precision is 100%, but if there were a total of 100 actual correct items, its recall is awful, at 1%. Similarly, if the model returned every item, its recall would be 100%, but its precision would be quite poor because most items should not be returned. The more items the model tries to return, the better its recall, but the harder it is to get them all right, so precision suffers. You can see from the diagram that 100% precision is driven by selecting only correct answers, while 100% recall results from selecting all correct answers, but that there are incorrect answers that can be false positives (they were selected when they should not have been) and false negatives (they were not selected when they should have been).

Precision measures the quality of the predictions that the model made. Recall measures the coverage by the model of all the actual correct predictions. Because precision and recall are essentially traded off one another, it's important that we are able to provide a single score that tells us when a model is more accurate than a person, or more accurate than another model, or more accurate than the previous version of the same model. That single measure is known as F-measure, and is a standard way to evaluate the accuracy of any model.

The way F-measures work is that they recognize that it is hard to improve both precision and recall at the same time, so they give credit for both. A typical F-measure, called an F1 Score, weighs precision and recall equally—because they are equally important to the user of the model. But it is possible to decide, for example, that it is worse to provide a wrong answer than to miss an answer, and to weight precision more heavily than recall. (For example, an F0.5 Score weighs precision twice as heavily as recall.) Conversely, you might decide that missing a correct answer is far worse than sometimes having incorrect answers, and decide to weight recall over precision. (For example, an F2 Score weighs recall twice as heavily as precision.) To conduct a proper evaluation, you must decide up front what the tradeoff is between precision and recall to calculate the most appropriate F-measures for your use case. Once you have decided that weighting, you now have an objective measurement that tells you how well your model is working.

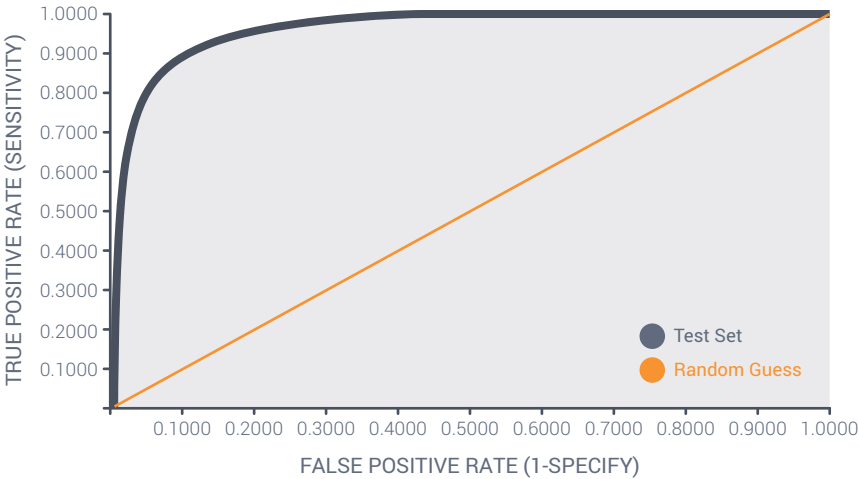
K

K is not in widespread use yet, having just recently been proposed in the research literature. If the appeal of F-measure is to sum up the evaluation of a model in a single number, K does that in a way that might be more effective than F-measure, because it can provide useful results even in situations when the test data is skewed to one outcome, rather than split evenly

K ranges between (-1, 1), with K = 1 indicating perfect performance, and K = -1 indicating completely incorrect performance. Random classifiers will have a K value of 0. A K value between 0.5 and 1 can generally be interpreted as indicating good model performance.

Unlike F-measures, K is robust and invariant to test set skew. For example, while testing a spam filter, where the only two outcomes are “spam” and “not spam,” if we have a test set containing only “spam” records, then there are no “not spam” records to find, meaning the recall measure is undefined. That means that F-measures are also undefined, so it can’t be used. K, on the other hand, still has a valid value in that case. Obviously having no relevant items in a test set is an unusual case, but it serves to illustrate that even in the most skewed situation, can be a useful measure when F-measure is not.

RECEIVER OPERATING CHARACTERISTIC (ROC) CURVE



Area Under the ROC Curve (ROC AUC)	0.9628
Youden's Index (J)	0.8018
Optimum Threshold (B)	0.6035
Threshold (B) 0.0 - 1.0	<div><div></div></div>
Threshold (B)	0.0281
True Positive Rate (TPR)	1.0000
False Positive Rate (FPR)	0.4568
True Negative Rate (TNR)	0.5432
False Negative Rate (FNR)	0.0000
Area Under the ROC Curve (ROC AUC)	0.7716





AREA UNDER THE (ROC) CURVE (AUC)

AUC is an excellent way to compare two models, but it requires some explanation.

First, we need to explain the metrics used in a Receiver Operating Characteristic (ROC) Curve:

- **True Positive Rate (TPR).** The number of true positives predicted by the model divided by the actual number of members of the positive class. TPR is just another name for recall, which we discussed above.
- **False Positive Rate (FPR).** The number of false positives divided by the number of actual negative cases.

A ROC curve is a graph of True Positive Rate vs. False Positive Rate across a range of threshold values. The Area Under the (ROC) Curve (AUC) is the area below that graphed curve.

Now that you know what it is, let's look at why it can be so useful:

- **It picks a winner when F-measure does not.** AUC helps measure whether the model will correctly identify a random positive example more often than it would incorrectly mark positive a random negative example. In other words, it depicts how likely it is that the model can effectively discern between the two cases. So, while two models that perform that same task might have nearly the same scores for precision, recall, and F-measure, the model with the higher AUC score is better.
- **It works even when the test data is badly skewed.** Like K, AUC is especially good in situations where your test data doesn't reflect reality, such as when almost all of your training data is relevant to the topic and only a few records are not relevant. You'd prefer to have a more balanced set of training data, but AUC can work even when your data is skewed.

AUC is an especially good way to measure the difference between the two versions of the same model, or two different models addressing the same task.

USING TRAINING SETS AND TESTING SETS

Everyone is familiar with the teacher who drills students relentlessly on exactly the questions that will be given on the test, and the students naturally do well on that test. This is derisively known as “teaching to the test”. But have the students really learned anything that isn’t on the test? Likely not.



Similarly, if you train a model on the same data that you use to test it, your model is likely to test well, but it is less likely to do well on new data that it has not seen before. So, it is critical to test on different data than you train with. This doesn’t mean that the data can’t come from the same source—it is fine for you to train on tweets and test on tweets. But it means that you don’t want to train and test on the same tweets.

The simplest way to do this is by using a hold-out set as shown in the diagram at right. As you code the raw data, you develop a large set of data known as gold standard data. That gold standard data is the ground truth—we know that we have the data and the correct answer. So, for example, for sentiment analysis for social data, we would have a list of tweets and its code of positive, negative, or neutral. We know that the data is accurate because we did the coding using inter-coder agreement, as described above. The next step is critical: we now separate the training set (which you can think of as the training sample) from the testing set (or testing sample). We “hold out” the testing set so that the model is never trained on that data. We train the model only on the training set. That is how to avoid teaching to the test.

If you plan on training the model, testing it, training it some more, and testing it some more, you might use a more complex approach to holding out data. Doing so requires careful planning of how to split up the gold standard data, which can be difficult to do, because there is no way to know how many times you might need to retrain and retest the data before you are getting the performance from the model that meets your needs. What you would like to do is to keep retraining and retesting until you are happy.

An even harder question is how much data do you need? If your gold standard data is imbalanced, such as sentiment data where there is far more neutral data than positive or negative data, you might need to amass much more data than if it is evenly split. The reason for that is that the model is recognizing patterns. If you have 5,000 coded records in your gold data, and there are 4,000 neutral records and just 500 each that are positive and negative, you will have a rich set of patterns for the model to recognize neutral items, but it might struggle to recognize positive and negative items without beefing up the number of those occurrences in your gold data.

But there is one more problem with hold-out sets that you might be wondering about. By just grabbing a slice of data and using that as the testing set, you are assuming that the data that you are using is representative of the entire set. You are assuming, essentially, that the data was randomly selected and that it is going to give you the same test result as any other data that you choose. The problem is that assumption is not necessarily true, and it has been shown that running the exact same model against the exact same gold data, but taking different slices of the data for testing can yield very different results.

Luckily, a more rigorous technique can be used that can dramatically reduce the deviation between multiple tests against the same data. Various known as cross-validation or striping, it offers a way for you to be highly confident that the evaluation you have performed is an adequate predictor of how your model will perform in real life.

ASSESSING THE MODEL AGAINST BUSINESS NEED

Let's assume you now have designed, validated, and deployed a high-performance model. Your F-measures are strong. Your model generalizes to different kinds of data and the quality of the predictions is high. The final area of analysis is how well the model meets the business objectives. Establishing KPIs on how well the model drives business value is a critical step to ensure models meet their purpose.

Often, there are key questions to answer:

- > **Can you reallocate budget now from more expensive initiatives?**
- > **Do the models increase automation and reduce the need for expensive human oversight?**
- > **Do they accelerate insights?**
- > **Are you getting better, more useful, and more actionable insights?**
- > **Is it driving adoption of the data across the organization?**

Establishing evaluation criteria on how these models drive business value is an important step to ensure the effective resourcing and scale out of this technology.

For even more detailed instructions on how to evaluate your social intelligence models, refer to a book chapter from Philip Resnik.



SUMMARY

As social data and related VoC data become increasingly adopted by leading organizations into critical functions, highly accurate social intelligence models have become a critical business need for many important use cases, such as market research and product development.

With so many vendors making promises of accuracy, and so many internal IT shops trying do-it-yourself data science, it's imperative that the user of social intelligence models understand the critical steps to building models that work. Whether you do it yourself or use a vendor product, you must inspect the development process to ensure that best practices are being followed:

You have the data. Different use cases require different data and different amounts of data. Make sure yours has sufficient data to train a model to the needed level of accuracy.

You trust your data. AI doesn't repeal the "garbage in, garbage out" rule. Take pains to ensure that your training data has been properly collected and accurately coded using stringent coding guidelines and inter-coder agreement techniques.

You continuously improve. AI is complex and it rarely delivers the best results on the first try. Follow an iterative process that constantly evaluates accuracy, classifies the critical errors, and roots them out.

You know how to keep score. No process will work if you can't be sure that you are evaluating the model's performance accurately. Use state-of-the-art model performance evaluation metrics, such as K and AUC, in addition to traditional F-measures, to ensure that you can tell when your model is improving and when it isn't, so you throw away the bad versions and build on the good ones.

Clearly understanding the performance of social intelligence models is essential. It requires rigorous coding of as much gold standard data as you can amass, a clear-eyed decision of the proper tradeoff between recall and precision, and a willingness to qualitatively evaluate the model and return to improving the training data to reach the desired level of accuracy. The framework described here is a critical step in ensuring the measurement is conducted objectively, effectively and consistently.

Don't trust your critical business decisions to a flawed process that risks artificial stupidity. Without adhering to best practices in developing and evaluating your models, you might find yourself having automated the wrong answer.

THE RISE OF “NO CODE” NLP: CONVERSUS.AI

All the processes and approaches covered here are all critically important for success. Weak links in any particular section can send model performance awry.

To help avoid this, NLP platforms can play an important role by automating many of these steps and allow even non-data scientists to rapidly build, validate and deploy effective models with confidence.

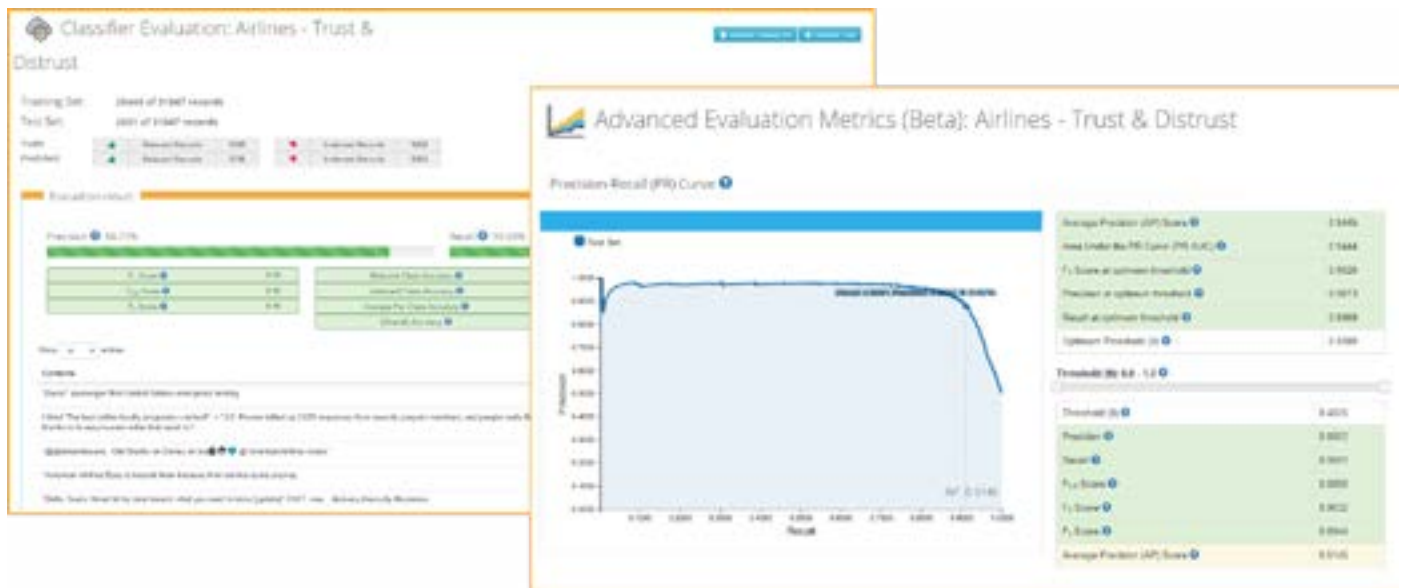
Conversus.AI is the leader in “no code” NLP platforms for this type of natural language processing. Provided by Converseon, the platform is already “intelligent,” having been rigorously “trained” over the last decade across more than 20 industries and more than a dozen languages. It was named “Top NLP” Platform by the 2019 AI Breakthrough Awards. With effective use, it’s models generally out perform individual humans and transform messy unstructured data into insight-rich “research-grade data.”

It is currently being used by a wide range of leading brands looking to level up their social and VoC language analysis. Users can generally build custom models in hours or days, rapidly accelerating the effectiveness of these models while reducing costs. It also provides access to a robust library of pre-built expert models for immediate.



Key capabilities of Conversus.AI include:

- > **Customize:** Custom and pre-built machine learning models specific to your industry, organization and business need, for a broad range of business needs in areas such as brand health, crisis analysis, trend spotting, customer care, customer experience and more. Provides target-level analysis and helps avoid inadvertent AI bias.
- > **Confidence:** Transparent automated model performance scoring and validation, and key features to align with ethical AI standards. The rigorous model performance approach described in this paper are largely automated in the system and also help model developers tune the model for optimal performance.



- > **Control:** It puts your subject matter experts in charge of your data and allows modifications to align with your frameworks and definitions. Through API connections, the models are programmatically integrated across a broad ecosystem of social listening, management and business intelligence platforms, such as Brandwatch, Linkfluence, Tableau and more, or your own internal data lakes.
- > **Cost Benefits:** Reduces model development, deployment and maintenance costs by up to 90% and increases performance by more than 50% over more standard model development approaches.

Data provided through Conversus.AI models have proven to have strong quantitative and predictive capabilities in multi-peer reviewed studies, especially in the areas of Brand Tracking, Trend Discovery, CX and "social NPS."

For a free demo please contact us at hello@converseon.com.

For nearly 15 years, Converseon has provided the full range of technology and consulting solutions to fully leverage insights from social and voice of customer data. These range from Conversus.AI "do it yourself" capabilities, to pre-built models, to full turnkey model development and maintenance.



CONCLUSION

The convergence of AI with natural language processing and social and voice of customer data is clearly a critically important development for customer-centric organizations. For many leading organizations, it is representing an entirely new generation of insight, including predictive insights, that are helping to transform brand guidance, reputation management, customer care, customer experience, trend discovery and market research more generally. However, as doing so effectively requires adopting clear and effective processes to “do it right.” When applied correctly, the power of this data + technology is clearly transformative and requires social listening experts, data scientists and market research experts (to name a few) who work with the data, to understand and apply best practice approaches to maximize impact while reducing the risks associated with misapplying the technology.

As industry analysts wrote in the Forrester Wave for Enterprise Social Listening Platforms, “Brands should be wary of over-exuberant AI promises.” And we agree. This next generation of social and customer feedback analysis can no longer simply rely on black box algorithms of unclear quality, but instead rely on transparent performance metrics and models that are designed following best practice methods such as those outlined here. We hope you find this guide of use and encourage feedback and thoughts as this guide evolves.

GLOSSARY

- **Accuracy Score.** The proportion of correct predictions divided by all items in the testing set. This is the “overall accuracy” score.
<https://developers.google.com/machine-learning/crash-course/glossary#accuracy>
- **Active Learning.** A machine learning training process whereby the system requests more training data for situations where the machine has less confidence in its predictions. For example, if a system asks a human expert to check all predictions below a certain confidence score, that can be rightly referred to as an active learning system.
[https://en.wikipedia.org/wiki/Active_learning_\(machine_learning\)](https://en.wikipedia.org/wiki/Active_learning_(machine_learning))
- **Annotation.** Data added to a record or document that categorizes or classifies it in some way. For example, a social media conversation could be annotated with the judgement of “positive” by a sentiment model.
<https://www.oreilly.com/library/view/natural-language-annotation/9781449332693/ch01.html>
- **Area Under (the ROC) Curve (AUC).** A means of comparing models to each other using a Receiving Operating Characteristic (ROC) Curve that is often superior to the tradition F-measure, because it works with skewed testing data and can sometimes indicate better discrimination among models
<https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>
- **Artificial Intelligence (AI).** A computer technique by which machines appear to be able to perform tasks previously thought to require human intelligence.
https://en.wikipedia.org/wiki/Artificial_intelligence
- **Average Per-Class Accuracy.** Also known as macro-average accuracy, the average of the accuracy scores for individual classes. This measure better accommodates class imbalance in the test set. For binary classification, it is equivalent to balanced accuracy.
<http://mvpa.blogspot.com/2015/12/balanced-accuracy-what-and-why.html>
- **Average Precision (AP) score.** A summary of a precision-recall curve as the weighted mean of precision achieved at each threshold.
[https://en.wikipedia.org/wiki/Evaluation_measures_\(information_retrieval\)#Average_precision](https://en.wikipedia.org/wiki/Evaluation_measures_(information_retrieval)#Average_precision)
- **Balanced Accuracy.** The average per-class accuracy when calculated for binary classification.
<http://mvpa.blogspot.com/2015/12/balanced-accuracy-what-and-why.html>
- **Big Data.** Techniques used to process massive amounts of data available within an organization or within the world, usually referred to so as to indicate the expanding nature of data and its importance in fueling artificial intelligence.
https://en.wikipedia.org/wiki/Big_data

-> **Binary Classification.** A process by which exactly two outcomes are distinguished, such as relevant vs. irrelevant.
https://en.wikipedia.org/wiki/Binary_classification

-> **Coder.** A human analyst who decides the correct answer for a particular record. For example, someone who creates the training data for a sentiment model must code each record with the correct answer in order to train the model. Sometimes also called a “labeler”.
[https://en.wikipedia.org/wiki/Coding_\(social_sciences\)](https://en.wikipedia.org/wiki/Coding_(social_sciences))

-> **Coding.** The act of deciding the correct answer for a particular record in the training data. For example, if the coder deems a record to convey positive sentiment according to the coding guidelines, the record should be coded as positive. Sometimes also called “labeling”.
[https://en.wikipedia.org/wiki/Coding_\(social_sciences\)](https://en.wikipedia.org/wiki/Coding_(social_sciences))

-> **Coding Guidelines.** A set of written instructions to aid the coder in making accurate decisions when coding the training data.
<https://getthematic.com/insights/coding-qualitative-data/>

-> **Confidence Level.** Also known as confidence score or confidence interval, a metric that indicates the degree to which the machine expects the prediction to be correct. High levels indicate that the machine is very confident about the prediction and low, the opposite. Sometimes, confidence scores are combined with a threshold to make a decision, such as referring predictions below a certain confidence level to humans for review, or tagging a document as belonging to a certain topic when the confidence level is above 80%.
https://en.wikipedia.org/wiki/Confidence_interval

-> **Confusion Matrix.** Used to suggest the prevalence of a particular type of error, a table that juxtaposes the correct and incorrect predictions by the model across each prediction value.
https://en.wikipedia.org/wiki/Confusion_matrix

-> **Cross-Validation.** Also called striping, a technique for testing a model that repeatedly selects different testing samples (or stripes) to evaluate the model, averaging the scores for each sample to provide an overall model performance metric that is more accurate than the score of any individual training sample.
[https://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics))

-> **Curation.** The act of manually selecting an item, such as assembling representative records for training data for machine learning.
https://en.wikipedia.org/wiki/Data_curation

-> **Data-Driven.** Operated by amassing data to bear on a problem, in contrast to using subjective human judgement or logically-defined rules.
<https://www.socure.com/blog/rules-are-meant-to-be-broken-machine-learning-vs.-rules-based-systems>

-> **Deep Learning.** A machine learning technique that generalizes training data so that a single training record might provide more value than you would expect based on its content. For example, Word2Vec is a deep learning facility that generalizes words so that any training record containing a word can also impart training about words that are similar to the actual word contained in the record. Using this kind of deep learning technique, a record containing the word queen can convey some learning to the related words royalty, ruler, monarch, king, princess, and prince, even though those words were not contained in the actual record.
https://en.wikipedia.org/wiki/Deep_learning

-> **Document-Level Annotation.** When a ML model makes a single prediction about an entire text “document”, the resulting output is called “document-level” annotation. Conversely, when a ML model makes multiple predictions about multiple spans of text within a document, it is called “entity-level” or “target-level” annotation.

-> **Expert System.** An early form of AI that used large numbers of rules to predict the answers to questions, sometimes successfully resembling the judgement of human experts for a particular question.
https://en.wikipedia.org/wiki/Expert_system

-> **False Negative.** An item incorrectly predicted to fall outside the class of desired items. For example, if a model is designed to predict which items fall within a certain topic, an item that is predicted to be outside the topic that actually lies within that topic is a false negative.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

-> **False Negative Rate (FNR).** The number of false negatives divided by true members of the positive class. FNR is opposite of recall.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

-> **False Positive.** An item incorrectly predicted to fall inside the class of desired items. For example, if a model is designed to predict which items fall within a certain topic, an item that is predicted to be within the topic that in truth lies outside that topic is a false positive.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

-> **False Positive Rate (FPR).** The number of false positives divided by the number of actual negative cases. False Positive Rate (FPR) is the opposite of specificity.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

-> **F-measure.** A measurement that summarizes the accuracy of a model by trading off precision vs. recall.
https://en.wikipedia.org/wiki/F1_score

-> **F0.5 Score.** A form of F-measure that gives twice as much weight to precision as opposed to recall.
https://en.wikipedia.org/wiki/F1_score

-> **F1 Score.** A form of F-measure that yields the harmonic mean of precision and recall, where they are weighed equally, reaching its optimal value at 1 (indicating perfect precision and recall) and its worst value at 0.
https://en.wikipedia.org/wiki/F1_score

-> **F2 Score.** A form of F-measure that gives twice as much weight to recall as opposed to precision.
https://en.wikipedia.org/wiki/F1_score

-> **Gold Standard Data.** The training data that has been carefully coded to be correct, using coding guidelines and inter-coder agreement, for example. This data is as close to being objectively correct as we can make it. The gold data is separated into testing samples and training samples.
[https://en.wikipedia.org/wiki/Gold_standard_\(test\)](https://en.wikipedia.org/wiki/Gold_standard_(test))
-> **Hold-Out Set.** Also known as a testing set, the subset of gold standard data that is used to test a model that has been trained on other data (the training set).
https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets#Holdout_method
-> **Human-in-the-Loop.** A type of technique where the predictions of machine learning models are augmented by human input. For example, a process where humans check some of the machine decisions can be correctly described as a human-in-the-loop system.
<https://appen.com/blog/human-in-the-loop/>
-> **Inter-Coder Agreement.** A technique where a single training record is coded by multiple coders in order to increase the accuracy of the training data. Records coded the same way by multiple coders are generally more accurate than those coded by just one.
https://en.wikipedia.org/wiki/Inter-rater_reliability
-> **J.** Also known as Youden's Index, a value that ranges from zero (completely incorrect) to one (perfect) that summarizes results at all points in a ROC Curve.
https://en.wikipedia.org/wiki/Youden%27s_J_statistic
-> **K.** A recent variant of balanced accuracy and Youden's Index (J) that ranges from -1 (completely incorrect) to 1 (perfect).
<https://drive.google.com/file/d/1Tz7cGuBzJpedJXx7Dkza2eLhAnnD2Yj/view>
-> **Machine Learning (ML).** A form of artificial intelligence by which a computer system trains a model that predicts the answer to a question.
https://en.wikipedia.org/wiki/Machine_learning
-> **Macro-Average Accuracy.** Also known as average per-class accuracy, the average of the accuracy scores for individual classes. This measure better accommodates class imbalance in the test set. For binary classification, it is equivalent to balanced accuracy.
<http://mvpa.blogspot.com/2015/12/balanced-accuracy-what-and-why.html>
-> **Model.** An artificial intelligence computer program typically produced by machine learning training that predicts the answer to a problem for each piece of data that it is presented.
https://en.wikipedia.org/wiki/Machine_learning
-> **Model Performance Metric.** Any measure of the accuracy of a model, such as F-measure or AUC.
<https://medium.com/usf-msds/choosing-the-right-metric-for-evaluating-machine-learning-models-part-2-86d5649a5428>
-> **Natural Language Processing (NLP).** Also known as text analytics, an artificial intelligence technique that extracts, parses, and analyzes text passages to create models of the text's meaning. For example, an NLP model could identify all proper nouns in a particular document and possibly segment the ones that are company names from places from people's names from others.
https://en.wikipedia.org/wiki/Natural_language_processing
-> **Optimum Threshold (θ).** The single threshold value where the vertical distance above the 45° line to the ROC curve, at its maximum, i.e. the threshold value where Youden's Index at its maximum is found on the ROC curve.
<https://machinelearningmastery.com/assessing-comparing-classifier-performance-roc-curves-2/>

-> **Per-Class Accuracy.** The score for individual classes in the testing set. This score indicates how the model is performing with respect to a particular class of interest.
https://en.wikipedia.org/wiki/Positive_and_negative_predictive_values

-> **Precision.** An indication of model performance where true positives are divided by the total number of positives predicted by the model.
<https://developers.google.com/machine-learning/crash-course/glossary#precision>

-> **Precision-Recall Curve.** A plot of precision vs. recall pairs computed over various classification thresholds. The ROC curve gives equal importance to the positive and negative class examples. When there are many incorrect predictions, the ROC will indicate poor model performance, but will not help in determining if the poor performance is due to incorrect negative or positive class prediction. The Precision-Recall curve can be more useful in cases where the error in positive class prediction is of particular interest.
<https://www.coursera.org/lecture/ml-classification/precision-recall-curve-rENu8>

-> **Recall.** Also known as sensitivity or True Positive Rate (TPR), the number of true positives predicted by the model divided by the actual positive items.
<https://developers.google.com/machine-learning/crash-course/glossary#recall>

-> **Receiving Operating Characteristic (ROC) Curve.** A plot of True Positive Rate (TPR) vs. False Positive Rate (FPR) which are computed over various classification thresholds. The TPR is on the y-axis in the ROC curve, and the FPR is on the x-axis in the ROC curve. The closer the curve is to the upper left, the better the model performance is.
<https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

-> **Rules-Based.** Operated by defining a large set of logical rules (e.g., "if this, then that") as opposed to using human judgement or being data-driven.
<https://www.socure.com/blog/rules-are-meant-to-be-broken-machine-learning-vs.-rules-based-systems>

-> **Semi-Supervised Machine Learning.** A form of machine learning which partially depends on training data that contains the correct answers for each record, and partially depends on input outside of the initial training data, a hybrid approach combining supervised and unsupervised machine learning.
https://en.wikipedia.org/wiki/Semi-supervised_learning

-> **Sensitivity.** Also known as recall or True Positive Rate (TPR), the number of true positives predicted by the model divided by the actual positive items.
<https://www.socure.com/blog/rules-are-meant-to-be-broken-machine-learning-vs.-rules-based-systems>

-> **Specificity.** Also known as True Negative Rate (TNR), the number of true negatives divided by the number of actual negative items.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

-> **Stripes.** A series of training samples taken from gold standard data, tested each in turn, as part of a cross validation process to evaluate the performance of a model.
[https://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics))

-> **Striping.** Also called cross-validation, a technique for testing a model that repeatedly selects different testing samples (or stripes) to evaluate the model, averaging the scores for each sample to provide an overall model performance metric that is more accurate than the score of any individual training sample.
[https://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics))

-> **Supervised Machine Learning.** A form of machine learning which depends on training data that contains the correct answers for each record, so as to train the model to predict such answers, in contrast to unsupervised machine learning, which requires no training data and is used to extract structure but not produce specific answers or perform specific tasks.
https://en.wikipedia.org/wiki/Supervised_learning

-> **Target-Level Annotation.** When a ML model makes multiple predictions about multiple spans of text within a document, it is called “entity-level” or “target-level” annotation. Conversely, when a ML model makes a single prediction about an entire text “document”, the resulting output is called “document-level” annotation.

-> **Testing Set.** Also known as a hold-out set, the subset of gold standard data that is used to test a model that has been trained on other data (the training set).
https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets#Holdout_method

-> **Text Analytics.** Also known as Natural Language Processing (NLP), an artificial intelligence technique that extracts, parses, and analyzes text passages to create models of the text’s meaning. For example, an NLP model could identify all proper nouns in a particular document and possibly segment the ones that are company names from places from people’s names from others.
https://en.wikipedia.org/wiki/Natural_language_processing

-> **Threshold.** A cut-off point used to make a decision. For example, predictions from models often include a confidence score as to how certain the machine is about the prediction—a threshold can be set so that predictions above a certain confidence are assumed to be correct and are used to automate a certain action rather than have humans perform it.
https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets#Holdout_method

-> **Training Data.** Data that is used to train a machine learning model containing data similar to what the model will see in production, as well as the correct outcome that the model should predict. By absorbing enough training data the machine learning model eventually “learns” to predict mostly correct outcomes for data it has not been trained on.
https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets#training_set

-> **Training Set.** The subset of gold standard data that is used to train a model that will be tested on other data (the testing set).
https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets#Holdout_method

-> **True Negative.** An item correctly predicted to fall outside the class of desired items. For example, if a model is designed to predict which items fall within a certain topic, an item that is predicted to be outside the topic that actually lies outside that topic is a true negative.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

-> **True Negative Rate (TNR).** Also known as specificity, the number of true negatives divided by the number of actual negative items.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

.....> **True Positive.** An item correctly predicted to fall inside the class of desired items. For example, if a model is designed to predict which items fall within a certain topic, an item that is predicted to be inside the topic that actually lies inside that topic is a true positive.
https://en.wikipedia.org/wiki/Sensitivity_and_specificity

.....> **True Positive Rate (TPR).** Also known as recall or sensitivity, the number of true positives predicted by the model divided by the actual number of positives.
<https://developers.google.com/machine-learning/crash-course/glossary#recall>

.....> **Unsupervised Machine Learning.** A form of machine learning which examines a class of data and clusters the data into similar groups or uncovers structure, in contrast to supervised machine learning, which requires training data and produces a model to perform a specific task.
https://en.wikipedia.org/wiki/Unsupervised_learning

.....> **Youden's Index.** Also known as J, a value that ranges from zero (useless) to one (perfect) that summarizes results at all points in a ROC Curve.
https://en.wikipedia.org/wiki/Youden%27s_J_statistic



