From Words to Distributions
A Probabilistic View of Language

Lado Turmanidze

October 1, 2025

Language as a Probabilistic System

- **Motivation:** Language is not just about grammatical correctness, but about how *likely* certain sequences of words are.
  - Humans expect some phrases more than others.
  - Probability models help predict, compress, and analyze language.

- **Example:** Suppose *Toyvelian* language only has the words $\{\text{"I"}, \text{"like"}, \text{"algebraists"}, \text{"logicians"}\}$.

$$P(\text{"I like logicians"}) = 0.5$$
$$P(\text{"I like algebraists"}) = 0.3$$
$$P(\text{"algebraists like logicians"}) = 0.15$$
$$P(\text{"logicians like algebraists"}) = 0.05$$

This probability distribution assigns likelihoods to different valid sentences.

Reminder: What is a Probability Distribution?

- A **random variable** $X : \Omega \to E$ is a measurable function from a **sample space** Ω (all possible outcomes) to a **measurable space** E (possible values we assign probabilities to).

- Examples:
  - **Die roll:**
    - Sample space: $\Omega = \{\omega_1, \ldots, \omega_6\}$ (underlying physical outcomes).
    - Measurable space: $E = \{1, 2, 3, 4, 5, 6\}$ (numbers shown).
    - Random variable: $X(\omega_i) = i$.
  - **Next word in a sentence:**
    - Sample space: $\Omega =$ all possible realizations of linguistic processes.
    - Measurable space: $E =$ vocabulary (finite set of words).
    - Random variable: $W(\omega) =$ the specific word produced.

- A **probability distribution** assigns likelihoods to the values in E and must satisfy: probabilities $\in [0, 1]$ and sum/integrate to 1.

Probability Mass Function (PMF)

- A **probability distribution** describes how likely each outcome of a random variable is.
- For **discrete** random variables, we use a **Probability Mass Function (PMF)**:
  - $p(x) = P(X = x)$ assigns probability to each discrete outcome.
  - Must satisfy $\sum_x p(x) = 1$.
  - Example: Fair die roll X: $p(2) = 1/6$.
- Relation to cumulative distribution function (CDF):

$$F(x) = P(X \leq x) = \sum_{t \leq x} p(t)$$

Probability Density Function (PDF)

- For **continuous** random variables, we use a **Probability Density Function (PDF)**:
  - A function $f(x)$ such that

$$P(a \le X \le b) = \int_a^b f(x)\,dx.$$

  - Must satisfy $\int_{-\infty}^{\infty} f(x)\,dx = 1$.
  - Example: $X \sim \mathcal{N}(0,1)$, $f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$.

- Relation to cumulative distribution function (CDF):

$$F(x) = P(X \le x) = \int_{-\infty}^{x} f(t)\,dt, \quad f(x) = \frac{d}{dx}F(x)$$

Joint Distributions

- A **joint distribution** describes probabilities of two or more random variables together.
- Notation:
  - Discrete: $P(X = x, Y = y)$
  - Continuous: $f_{X,Y}(x, y)$
- Example:
  - In language: joint probability of two words

$$P("\text{I}", "\text{like}") = 0.4$$

- Marginals can be recovered by summing/integrating:

$$P(X = x) = \sum_y P(X = x, Y = y) \quad \text{or} \quad f_X(x) = \int f_{X,Y}(x, y)\, dy$$

Formal Language

- **Alphabet:** A finite set $A = \{a_1, \ldots, a_n\}$ with $|A| = n$.
- **String:** A word $b = b_1 b_2 \ldots b_\ell$ where $\forall i \in [1, \ell],\ b_i \in A$.
- **Length:** $|b| = \ell$.
- **Empty word:** Denoted ε, with $a \circ \varepsilon = \varepsilon \circ a = a$.
- **Concatenation:** If $b = b_1 \ldots b_r$ and $c = c_1 \ldots c_s$, then

$$b \circ c = b_1 \ldots b_r c_1 \ldots c_s.$$

- **Kleene closure:**

$$A^* = \bigcup_{i=0}^{\infty} A^i = A^+ \cup \{\varepsilon\},$$

where $A^i = \{w : |w| = i \wedge \forall j \in [1, i],\ w_j \in A\}$ and

$$A^+ = \bigcup_{i=1}^{\infty} A^i.$$

- **Language:** Any subset $L \subseteq A^*$.

Probabilistic Language Model

- A **probabilistic language model** is a probability distribution

  $$P : A^* \to [0,1]$$

  such that

  $$\sum_{w \in A^*} P(w) = 1.$$

- For a string $w = w_1 w_2 \ldots w_n \in A^*$, the model assigns probability

  $$P(w) = P(w_1, w_2, \ldots, w_n).$$

- Intuitively: a language is not only a set of well-formed strings, but also a *distribution* that determines how likely each string is.

Why Compare Language Models?

- Suppose we have:
  - $P \leftarrow$ true distribution of sentences in a language (from a corpus - a large collection of text)
  - $Q \leftarrow$ a model-generated distribution (e.g., an LLM)
- We want to know:
  - How well does Q capture the “patterns” of real language?
  - Which model better predicts or generates natural text?
- Intuition:
  - If Q is very different from P, generated sentences may be unnatural or unlikely.
  - If Q is similar to P, model outputs are closer to human-like language.

Entropy: Surprise of a Single String

- Introduced by Claude Shannon (1948) **information theory**.
- The **surprise** of seeing a string w in a distribution P is

$$I(w) = -\log P(w)$$

- Rare strings ($P(w)$ small) are more surprising $\implies$ high $I(w)$.
- Common strings ($P(w)$ large) are less surprising $\implies$ low $I(w)$.
- Logarithm ensures additive behavior for independent events: $I(w_1 \circ w_2) = I(w_1) + I(w_2)$.

Entropy: Average Uncertainty of a Language

- Entropy $H(P)$ measures the **expected surprise** when sampling from P:

$$H(P) = \sum_{w \in A^*} P(w) \cdot I(w) = -\sum_{w \in A^*} P(w) \log P(w)$$

- Intuition:
  - High entropy $\implies$ many plausible sentences $\implies$ more uncertainty.
  - Low entropy $\implies$ few dominant sentences $\implies$ more predictability.

KL Divergence: Comparing Language Models

- Suppose we have two probabilistic models over the same language:
  - $P \leftarrow$ true distribution from corpus (real language)
  - $Q \leftarrow$ model-generated distribution (LLM output)
- **KL divergence** measures how "different" Q is from P:

$$D_{KL}(P \,||\, Q) = \sum_{w \in A^*} P(w) \log \frac{P(w)}{Q(w)}$$

- Intuition:
  - $D_{KL}(P||Q) = 0$ if $Q \equiv P$
  - Small $D_{KL} \implies P$ and Q are relatively close
  - Large $D_{KL} \implies Q$ assigns probability very differently from $P \implies$ more "surprise"
  - Think of it as "extra surprise when using Q instead of P"

KL Divergence in Action and Cross-Entropy

- Suppose our model Q predicts:

$$Q(\text{"I like logicians"}) = 0.5$$
$$Q(\text{"I like algebraists"}) = 0.3$$
$$Q(\text{"algebraists like logicians"}) = 0.15$$
$$Q(\text{"logicians like algebraists"}) = 0.05$$

- **Connection to cross-entropy:**

$$H(P,Q) = -\sum_w P(w)\log Q(w),\ D_{KL}(P||Q) = H(P,Q) - H(P)$$

  - Cross-entropy = average surprise using model Q instead of the true P
  - Minimizing cross-entropy in training LLMs is equivalent to minimizing KL divergence to the true language distribution

Training Language Models with Cross-Entropy

- During LLM training, we have:
  - $P \leftarrow$ empirical distribution from training corpus
  - $Q_\theta \leftarrow$ model distribution (depends on parameters θ)
- Training objective: minimize

$$\mathcal{L}(\theta) = H(P, Q_\theta) = -\sum_{w} P(w) \log Q_\theta(w)$$

- Minimizing cross-entropy $\iff$ minimizing KL divergence to true language distribution.
- Intuition: model learns to assign high probability to real sentences.

Perplexity: Definition and Intuition

- **Definition:** Perplexity is the exponentiated entropy of a distribution:

$$PP(P) = 2^{H(P)} \quad \text{or} \quad PP(P, Q) = 2^{H(P,Q)} \text{ for a model } Q$$

- Intuition:
  - Perplexity measures how "confused" a model is when predicting text.
  - Low perplexity $\implies$ model predicts likely strings well (confident)
  - High perplexity $\implies$ model assigns low probability to likely strings $\implies$ more uncertain

Perplexity: Example of Unnatural Repetition

- Suppose a model generates the sentence:

  `"I like like like logicians"`

  instead of the natural sentence:

  `"I like logicians"`

- True Toyvelian probabilities:

$$P(\text{"I like logicians"}) = 0.5,$$
$$P(\text{"I like algebraists"}) = 0.3, \ldots$$

- The model over-predicts repeated "like":

$$Q(\text{"I like like like logicians"}) = 0.4,$$
$$Q(\text{"I like logicians"}) = 0.05, \ldots$$

Perplexity: Sequence-Level Intuition and Caveats

- **Perplexity over a sequence:** For a sentence $w_1 w_2 \dots w_n$:

$$PP(P, Q) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n} \log Q(w_i)\right)$$

  - Measures the *average surprise per word*.
  - Low perplexity $\implies$ model predicts likely words well.
  - High perplexity $\implies$ model is “confused” or deviates from true distribution.
- **Caveats / Limitations:**
  - Does not capture semantic correctness or coherence.
  - Models can have low perplexity yet generate repetitive or unnatural text.
    - Toyvelian: "I like like like logicians"
    - English: "The cat sat on the mat the cat sat on the mat"
  - Useful as a statistical measure, but not a perfect evaluation of language quality.

Questions?