

TRANSDUCERS ETC

LINGUISTICS 684.02

GOALS

- IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.
- DETAILED CORRECTNESS IS ALL.
- WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT
- AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE

GOALS

- IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.
- DETAILED CORRECTNESS IS ALL.
- WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT
- AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE



John eats pizza

GOALS

IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.

DETAILED CORRECTNESS IS ALL.

WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT

AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE

✓	John eats pizza
✗	The pizza that John eats are big

GOALS

- IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.
- DETAILED CORRECTNESS IS ALL.
- WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT
- AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE

✓	John eats pizza
✗	The pizza that John eats are big
✓	The pizza that John eats is big

GOALS

- IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.
- DETAILED CORRECTNESS IS ALL.
- WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT
- AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE

✓	John eats pizza
✗	The pizza that John eats are big
✓	The pizza that John eats is big
✗	The pizzas that John eats is big

GOALS

- IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.
- DETAILED CORRECTNESS IS ALL.
- WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT
- AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE

✓	John eats pizza
✗	The pizza that John eats are big
✓	The pizza that John eats is big
✗	The pizzas that John eats is big
✓	The pizzas that John eats are big

GOALS

- IN SYNTACTIC THEORY, THE GOAL IS TO CREATE MODELS THAT MIRROR HUMAN JUDGEMENTS.
- DETAILED CORRECTNESS IS ALL.
- WE CAN ASSUME IDEALIZED, ERROR-FREE INPUT
- AND THERE ARE ASPECTS THAT WE DO NOT CHOOSE

✓	John eats pizza
✗	The pizza that John eats are big
✓	The pizza that John eats is big
✗	The pizzas that John eats is big
✓	The pizzas that John eats are big
✓	The pizzas that John eats are inclement

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

0.8

John eats pizza

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

0.8	John eats pizza
0.4	The pizza that John eats are big

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

0.8	John eats pizza
0.4	The pizza that John eats are big
0.65	The pizza that John eats is big

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

0.8	John eats pizza
0.4	The pizza that John eats are big
0.65	The pizza that John eats is big
0.2	The pizzas that John eats is big

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

0.8	John eats pizza
0.4	The pizza that John eats are big
0.65	The pizza that John eats is big
0.2	The pizzas that John eats is big
0.9	The pizzas that John eats are big

GOALS

- IN NLP, THE GOAL IS TO CREATE MODELS THAT ARE USEFUL.
- EFFICIENCY IS A MAJOR CONCERN
- WE WILL ENCOUNTER ERRORS OF ALL SORTS
- WE MAY BE PREPARED TO TRADE DETAIL FOR ROBUSTNESS.

0.8	John eats pizza
0.4	The pizza that John eats are big
0.65	The pizza that John eats is big
0.2	The pizzas that John eats is big
0.9	The pizzas that John eats are big
0.7	The pizzas that John eats are inclement


GOALS

GOALS

eats OBJ pizza: eats SUBJ John

John eats pizza

GOALS

eats OBJ pizza: eats SUBJ John		John eats pizza
pizza(p) & eats(j,p) & john(j)		The pizza that John eats are big



GOALS

eats OBJ pizza: eats SUBJ John		John eats pizza
pizza(p) & eats(j,p) & john(j)		The pizza that John eats are big
pizza(p) & eats(j,p) & john(j) & big(p)		The pizza that John eats is big

GOALS

eats OBJ pizza: eats SUBJ John		John eats pizza
pizza(p) & eats(j,p) & john(j)		The pizza that John eats are big
pizza(p) & eats(j,p) & john(j) & big(p)		The pizza that John eats is big
pizza(ps) & eats(j,p) & john(j) & big(p) & subset(p,ps)		The pizzas that John eats is big

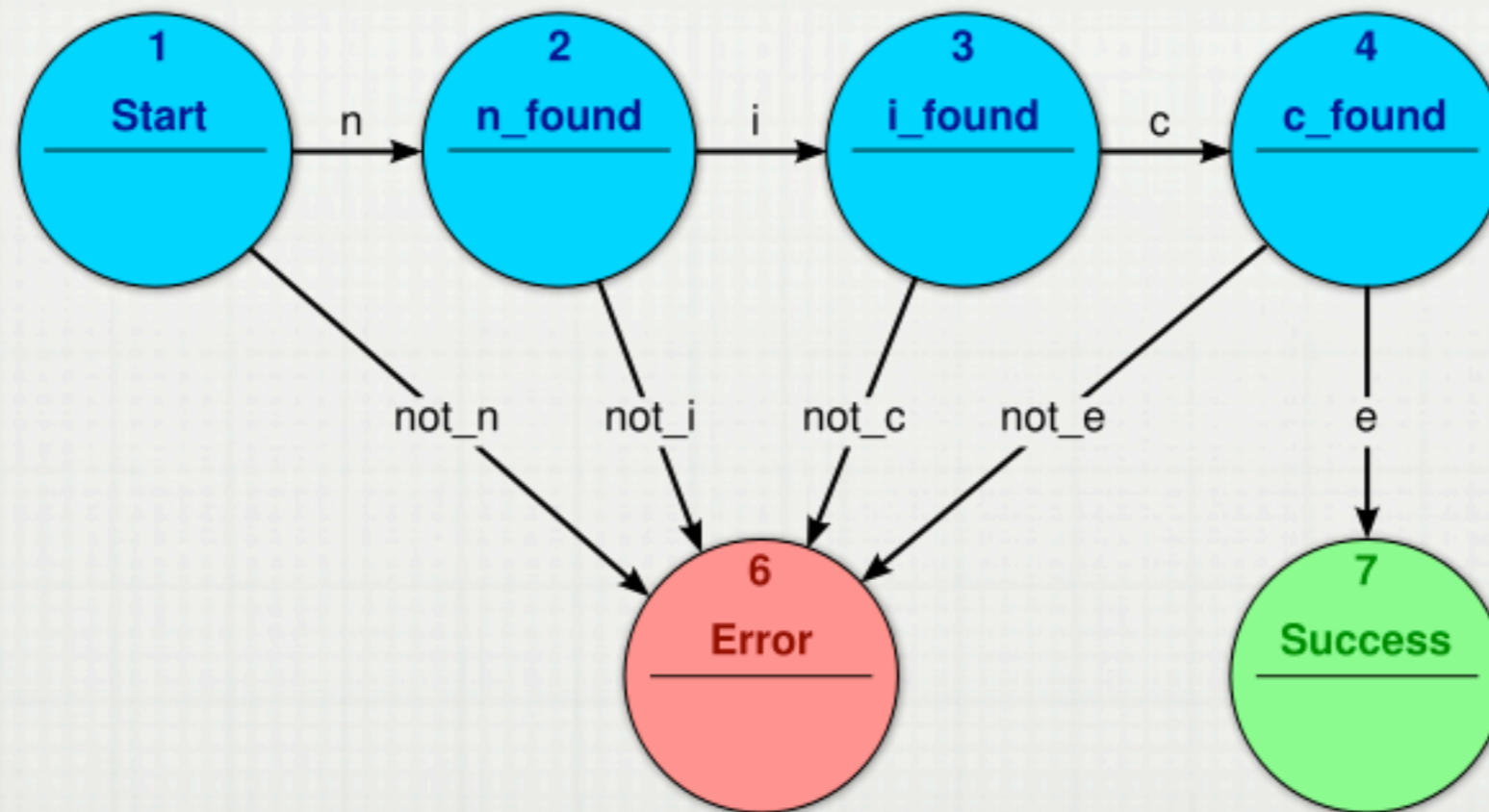
GOALS

eats OBJ pizza: eats SUBJ John		John eats pizza
pizza(p) & eats(j,p) & john(j)		The pizza that John eats are big
pizza(p) & eats(j,p) & john(j) & big(p)		The pizza that John eats is big
pizza(ps) & eats(j,p) & john(j) & big(p) & subset(p,ps)		The pizzas that John eats is big
pizza(ps) & eats(j,p) & john(j) & big(p) & subset(p,ps)		The pizzas that John eats are big

GOALS

eats OBJ pizza: eats SUBJ John		John eats pizza
pizza(p) & eats(j,p) & john(j)		The pizza that John eats are big
pizza(p) & eats(j,p) & john(j) & big(p)		The pizza that John eats is big
pizza(ps) & eats(j,p) & john(j) & big(p) & subset(p,ps)		The pizzas that John eats is big
pizza(ps) & eats(j,p) & john(j) & big(p) & subset(p,ps)		The pizzas that John eats are big
pizza(ps) & eats(j,p) & john(j) & inclement(p) & subset(p,ps)		The pizzas that John eats are inclement

FINITE STATE AUTOMATA



- SIMPLE IDEA. THE COMPUTATION MAY BE IN ANY OF SEVERAL STATES
- TRANSITIONS MOVE SYSTEM FROM ONE STATE TO ANOTHER.

FINITE STATE AUTOMATA

- A FINITE STATE AUTOMATON CONSISTS OF
 - A SET OF STATES
 - A SET OF ARCS JOINING THESE STATES, LABELED WITH SYMBOLS CHOSEN FROM SOME ALPHABET
 - AN INDICATION OF WHICH STATES ARE LEGAL START AND FINISH POINTS.

ALPHABETS

- MATHEMATICALLY, IT MAKES ALMOST NO DIFFERENCE WHAT SIZE THE ALPHABET IS.
- IN COMPUTER APPLICATIONS, ALPHABETS CAN BE DIFFERENT SIZES, INCLUDING INFINITE, AND THE CHOICE OF ALPHABET AFFECTS FEASIBILITY.
- TYPICAL ALPHABETS ARE LETTERS (FEW) OR WORDS (MANY).

ALPHABETS

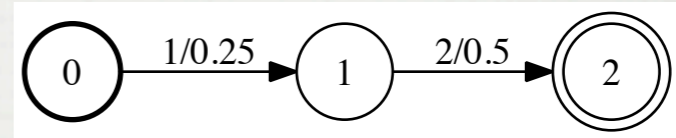
- INTUITIVELY, WE EXPECT TO SEE ALMOST ALL THE POSSIBLE LETTERS IN A FAIRLY SHORT SPAN OF TEXT (BUT BEWARE 'W' IN FRENCH)
- WE DO NOT EXPECT TO SEE ALL POSSIBLE WORDS.
- FOR ENGLISH, WE CAN LIST THE POSSIBLE FORMS OF THE VERB "MOVE". IN SLAVIC LANGUAGES, FOR EXAMPLE, THERE ARE MANY MORE WORD FORMS, EVEN FOR JUST ONE WORD.

ALPHABETS

- HANDLING PREVIOUSLY UNSEEN ITEMS IS A COLOSSAL RECURRING PROBLEM IN NLP.
- WON'T OFTEN ARISE FOR SINGLE LETTERS OF A 26 LETTER ALPHABET.
- WILL OFTEN ARISE FOR ALMOST ANYTHING ELSE,

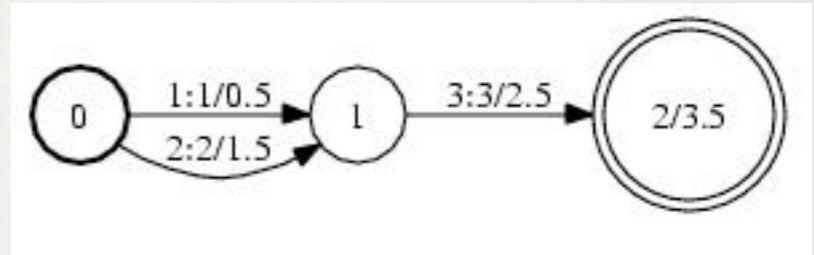
WEIGHTED AUTOMATA

- BOTH SYMBOLS AND WEIGHTS ON THE ARCS.
- OFTEN, BUT NOT ALWAYS, WEIGHTS ARE PROBABILITIES



FINITE STATE TRANSDUCERS

- BATTLE TESTED TOOLS FOR PROVIDING "ANALYSIS" THAT IS SOMEHOW CONNECTED TO THE INPUT
- CONCEPTS THAT ARE POWERFUL ENOUGH TO GET THINGS GOING.
- NOT ABLE TO SAY EVERYTHING THAT ONE MIGHT WANT TO.
- KEY OPERATION IS COMPOSITION. THIS IS A GENERALIZATION OF INTERSECTION OF FSAS



HAS AN INPUT
SIDE AND AN
OUTPUT SIDE, SO
CAN EXPRESS
RELATIONSHIP
BETWEEN
STRINGS

SUM AND PRODUCT

□ FREQUENTLY, WE ARE INTERESTED IN FINDING THE PROBABILITY OF A SEQUENCE OF TRANSITIONS.

□ IF SO, WE ARE INTERESTED IN ASSIGNING PROBABILITIES TO EACH STEP IN THE SEQUENCE.

$$p(x_1 \dots x_n) = \prod_i^n p(x_i | x_{i-1})$$

LOGARITHMS

$$\log p(x_1 \dots x_n) = \sum_i^n \log p(x_i | x_{i-1})$$

- LOGARITHMS ARE OFTEN USED AS WEIGHTS.
- A KEY FACT ABOUT LOGS IS THAT PRODUCTS OF PROBABILITIES CAN BE IMPLEMENTED AS SUMS OF LOG PROBABILITIES.

LOGARITHMS

$$\log p(x_1 \dots x_n) = \sum_i^n \log p(x_i | x_{i-1})$$

- ALSO, LOGARITHMS GET SMALL SLOWER, SO YOU RUN INTO THE NUMERICAL LIMITS OF COMPUTER REPRESENTATIONS OF NUMBERS SLOWER.
- AND LOGARITHMS ARE ORDER-PRESERVING, SO THE MAXIMUM OF THE LOG OF A QUANTITY IS IN THE SAME PLACE AS THE MAXIMUM OF THE UNDERLYING QUANTITY

BASE OF A LOGARITHM

$$\log e^x = x \quad \text{NATURAL LOG}$$

$$\log_{10} 10^x = x \quad \text{LOG BASE 10}$$

$$\log_2 2^x = x \quad \text{LOG BASE 2}$$

LOGS

- LOGS BASE 2 ARE COMMON IN INFORMATION THEORY, BECAUSE A DIFFERENCE OF 1 IN LOG BASE 2 CORRESPONDS TO A CHOICE PROBLEM THAT CAN BE RESOLVED BY KNOWING THE OUTCOME OF A SINGLE FAIR COIN FLIP. THIS AMOUNT OF DIFFERENCE IN INFORMATION IS CALLED A BIT.

LOGS

- LOGS BASE e (NATURAL LOGS) ARE COMMON IN MATH, BECAUSE SOME CALCULATIONS COME OUT SMOOTHLY IF THIS IS THE BASE.
- A DIFFERENCE OF ONE UNIT IN NATURAL LOGS IS CALLED A NAT. SOME CALCULATIONS (OFTEN STATISTICAL TESTS) NEED NATS.

LOGS

- LOGS BASE 10 CONVENIENTLY LINE UP WITH OUR NUMBER SYSTEM, SO $\log 10$ IS 1 AND $\log 100$ IS 2
- A DIFFERENCE OF 1 UNIT OF LOG BASE 10 IS A BEL
- BUT THIS CHOICE WOULD MAKE NO SENSE TO THE SIMPSONS, OR TO TWELVE-FINGERED ALIENS.

MAX AND ARGMAX

$$\max_x \prod_i p(x_i | x_0 \dots x_{i-1})$$

$$\operatorname{argmax}_x \prod_i p(x_i | x_0 \dots x_{i-1})$$

MEDICAL DIAGNOSIS

THE DOCTORS PROBLEM: $P(S,C)$ VS $P(S,P)$

CAUSAL INFORMATION: $P(S|C) = P(S|P) = 1$

BASE RATES: $P(P) = 10^{-6}$ $P(C) = 0.25$ $P(S) = 0.33$

WANTS: $P(C|S)$ AND $P(P|S)$

BEING BAYESIAN

POSTERIOR \propto PRIOR \times LIKELIHOOD

$$P(L|X) \propto P(L) \times P(X|L)$$

GRAMMAR INFERENCE = PRIOR BELIEFS ABOUT POSSIBLE
GRAMMARS \times LANGUAGE MODEL

LINGUISTIC DIAGNOSIS: LANGUAGE IDENTIFICATION

E PREEBAS BIOQUIMICA

MAN IMMUNODEFICIENCY

FAITS SE SONT PRODUI

BAD TECHNIQUES FOR LANGUAGE IDENTIFICATION

- COMMON WORDS: NEEDS LONGISH TEST DATA.
- PROPER NAMES: ESPECIALLY BAD, SINCE VERY NUMEROUS, AND HIGHLY LIKELY TO APPEAR IN DOCUMENTS IN ANOTHER LANGUAGE.
- UNIQUE LETTER PAIRS OR SEQUENCES: OK SOMETIMES, BUT DOESN'T WEIGHT EVIDENCE BY FREQUENCY

TUTORIAL PAPER BY TED DUNNING

DUNNING ASKS THE FOLLOWING QUESTIONS:

Q: HOW SIMPLE CAN THE PROGRAM BE?

A: SMALL PROGRAM BASED ON STATISTICAL PRINCIPLES

Q: WHAT DOES IT NEED TO LEARN?

A: NO HAND-CODED LINGUISTIC KNOWLEDGE IS NEEDED.
ONLY TRAINING DATA PLUS THE ASSUMPTION THAT
TEXTS ARE SOMEHOW MADE OF BYTES.

Q: HOW MUCH TRAINING DATA NEEDED?

MORE QUESTIONS

Q: HOW MUCH TEST DATA?

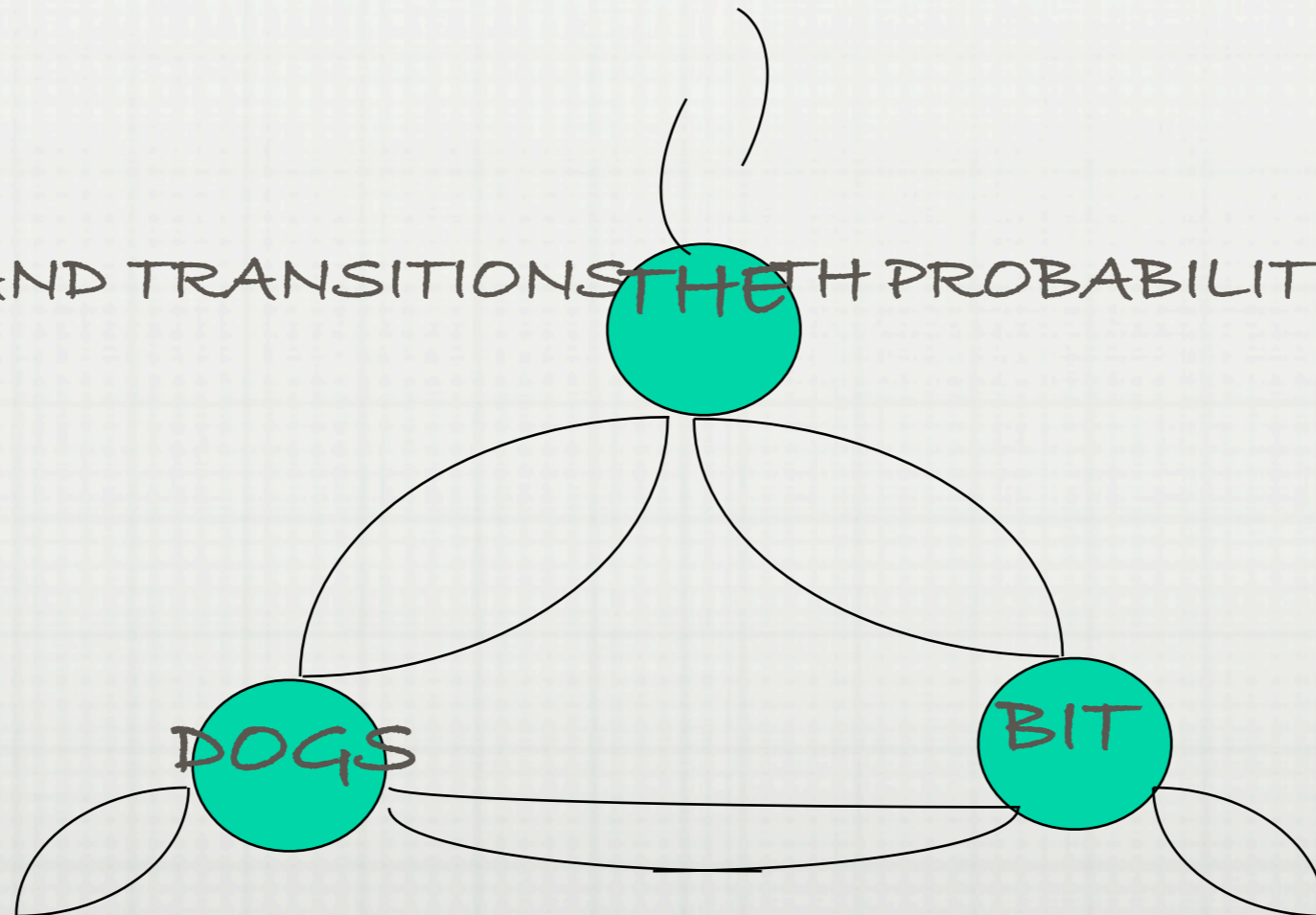
A: 10 CHARACTERS WORK, 500 CHARACTERS VERY WELL.

Q: CAN IT GENERALISE?

A: IF TRAINED ON FRENCH, ENGLISH AND SPANISH, THINKS GERMAN IS ENGLISH.

MARKOV MODELS

- STATES AND TRANSITIONS (WITH PROBABILITIES)



TABULAR FORM OF MARKOV MODELS

- TRANSITION MATRIX (A)

THE DOGS BIT

THE 0.33 0.33 0.33

DOGS 0.33 0.33 0.33

BIT 0.33 0.33 0.33

- START WITH INITIAL PROBABILITIES P(0)

THE 0.7

DOGS 0.2

BIT 0.1

USING MARKOV MODELS

- CHOOSE INITIAL STATE FROM $\underline{p}(0)$
- CHOOSE TRANSITION FROM RELEVANT ROW OF \underline{A}
- REPEAT AD LIB
- SIMPLE PROBABILISTIC GENERATOR FOR SEQUENCES OF WORDS. YIELDS SEQUENCE AND ITS PROBABILITY.
- SAME CAN BE DONE WITH CHARACTERS, GIVING A PROBABILISTIC GENERATOR FOR CHARACTER STRINGS.

HIGHER ORDER MARKOV MODELS

- IF YOU WANT TO TAKE ACCOUNT OF MORE CONTEXT, YOU CAN RE-DRAW THE MACHINE SO THAT EACH STATE IS LABELLED WITH A LONGER FRAGMENT OF THE CONTEXT.

RANDOMLY GENERATED STRINGS FROM MARKOV MODELS

0 HM 1 IMUANDO~DOC NI LEOTLS AIQE1PDT6CF TLC.TEONTCTRRDSXO~ES LOO OIL3S

1 ~ A MESTON S OFLAS N, ~ 2 NIKEXIHOMANOTRMO S,~125 0 3 1 35 FO THERE

2 S IST SES ANATP SUP SURES ALIHOWS RAAIAL ON TERLIKETICANY OF PRELLY

3 APPRODUCTION WHERE. IF THE LINERAL WATE PROBABILITY THE OR LIKELIHOOD

4 SUMED NORMAL OF THE NORMAL DISTRIBUTION. GALE, CHURCH,WILLINGS. THIS

5 ~K SUB 1} SUP {N-K} .EN WHERE THAN ROUGHLY 5. THIS AGREEMENTED BY TH

6 THESE MEAN IS NOT WORDS CAN BE SAID TO SPECIFY APPEAR MCDONALD. 1989

BAYESIAN DECISION RULES

- SPANISH AND ENGLISH AS DISEASES CAUSING THE SYMPTOM "IMMUNOTECHNO"
- COMPARE $P(\text{IMMUNOTECHNO}, \text{SPANISH})$ WITH $P(\text{IMMUNOTECHNO}, \text{ENGLISH})$
- IF WE WANT WE CAN COMPARE $P(\text{IMMUNOTECHNO} | \text{SPANISH})P(\text{SPANISH})$ WITH $P(\text{IMMUNOTECHNO} | \text{ENGLISH})P(\text{ENGLISH})$

BAYESIAN DECISION RULES2

- ASSUME THAT $P(\text{SPANISH}) = P(\text{ENGLISH})$.
- SO COMPARE $P(\text{IMMUNOTECHNO} | \text{SPANISH})$ WITH $P(\text{IMMUNOTECHNO} | \text{ENGLISH})$
- TO DO THIS, WE NEED A MODEL WHICH WILL GENERATE CHARACTER STRINGS FROM DATA.
- A CHARACTER-BASED MARKOV MODEL IS WHAT IS NEEDED

CHARACTER MARKOV MODELS

- $P(S^0 \dots S^N) = P(S^0)P(S^1 | S^0)P(S^2 | S^0, S^1)P(S^3 | S^0, S^1, S^2) \dots$
 $P(S^N | S^0 \dots S^{N-1})$
- THE ABOVE IS EXACT, BUT IMPRACTICAL, BECAUSE WE DON'T HAVE SUFFICIENT DATA FOR RELIABLE ESTIMATES.
- APPROXIMATE WITH $P^*(S^0 \dots S^N) = P(S^0)P(S^1 | S^0)P(S^2 | S^0, S^1)P(S^3 | S^1, S^2) \dots P(S^N | S^{N-2}, S^{N-1})$

CHARACTER MARKOV MODELS

- THAT WAS FOR CHARACTER-LEVEL MARKOV MODELS OF ORDER 2, WHICH CORRESPONDS TO A TRIGRAM MODEL. WORD-LEVEL TRIGRAM MODELS ARE COMMON.
- FOR THE LANGUAGE IDENTIFICATION APPLICATION WE ACTUALLY USE CHARACTER 4-GRAMS.

PROBABILITY ESTIMATION

- WE NEED TRANSITION PROBABILITIES $P(S_4 | S_1, S_2, S_3)$
- OBVIOUS WAY TO PROCEED IS TO COLLECT $|S_1, S_2, S_3, S_4|$ AND $|S_1, S_2, S_3|$ THEN DIVIDE.
- THIS GIVES ZERO IF $|S_1, S_2, S_3, S_4| = 0$, UNDEFINED IF $|S_1, S_2, S_3| = 0$

PROBABILITY ESTIMATION 2

- IT'S A GREAT MODEL FOR THE TRAINING SET, BUT FAILS CATASTROPHICALLY IN THE REAL WORLD.
- THERE MAY BE $k+1$ -GRAMS IN THE TEST DATA WHICH ARE ABSENT FROM THE TRAINING DATA.

PROBABILITY ESTIMATION 3

- BY BAD LUCK MAY BE A $k+1$ GRAM IN THE TRAINING DATA FOR ONE LANGUAGE, EVEN THOUGH IT IS IN FACT RARE IN ALL THE LANGUAGES.
- IF THIS HAPPENS, ALL STRINGS CONTAINING THAT $k+1$ -GRAM WILL BE JUDGED TO BE FROM THAT LANGUAGE, BECAUSE ALL THE OTHERS WILL BE 0.
- MAXIMUM LIKELIHOOD ESTIMATOR IS TOO BRITTLE.

SMOOTHING

- $P((s_4 | s_1, s_2, s_3, \text{LANG})) = \frac{(|s_1, s_2, s_3, s_4| + 1)}{(|s_1, s_2, s_3| + M)}$
- THIS IS MORE STABLE, CAN BE SEEN AS MIXING IN A SMALL MEASURE OF UNIFORM DISTRIBUTION INTO THE EMPIRICAL DATA
- *DISCOUNTS* THE DATA, BUT ALSO PREVENTS ZEROS
- LAPLACE'S M-ESTIMATE

RESULTS

- IN A BINARY CHOICE BETWEEN ENGLISH AND SPANISH STRINGS DRAWN FROM A BILINGUAL CORPUS, AN ACCURACY OF 92% FROM 20 BYTES OF TEST DATA AND 50KBYTES OF TRAINING DATA,
- IMPROVING TO ABOUT 99.9% WHEN 500 BYTES OF TEST DATA ARE ALLOWED.