Information-theoretic locality properties of natural language

Richard Futrell
Department of Language Science
Department of Computer Science
University of California, Irvine

@rljfutrell
rfutrell@uci.edu

Quantitative Syntax 2019
2019-08-26

Efficiency Hypothesis

Efficiency Hypothesis

- Each human language is a solution to the problem of **maximally efficient communication**…

Efficiency Hypothesis

- Each human language is a solution to the problem of **maximally efficient communication**…
- *subject to* fixed **human information processing constraints**.

Efficiency Hypothesis

- Each human language is a solution to the problem of **maximally efficient communication**…
- *subject to* fixed **human information processing constraints**.

- **Efficiency Hypothesis**: Languages are optimized so that messages we want to express are easy to produce and comprehend accurately.
  (Zipf, 1949; Hockett, 1960; Slobin, 1973; Givón, 1991, 1992; Hawkins, 1994, 2004, 2014; Christiansen & Chater, 2008; Jaeger & Tily, 2011; Fedzechkina et al., 2012; MacDonald, 2013)

Efficiency Hypothesis

- **Efficiency Hypothesis**: Languages are optimized so that messages we want to express are easy to produce and comprehend accurately.
(Zipf, 1949; Hockett, 1960; Slobin, 1973; Givón, 1991, 1992; Hawkins, 1994, 2004, 2014; Christiansen & Chater, 2008; Jaeger & Tily, 2011; Fedzechkina et al., 2012; MacDonald, 2013)

Efficiency Hypothesis

- **Efficiency Hypothesis**: Languages are optimized so that messages we want to express are easy to produce and comprehend accurately.
(Zipf, 1949; Hockett, 1960; Slobin, 1973; Givón, 1991, 1992; Hawkins, 1994, 2004, 2014; Christiansen & Chater, 2008; Jaeger & Tily, 2011; Fedzechkina et al., 2012; MacDonald, 2013)

- **Mathematical formalization**: human languages are **solutions** to a **constrained optimization problem** describing communication subject to cognitive constraints.
  - So, what is the objective function that human languages optimize?

Information-Theoretic Models of Natural Language

Information-Theoretic Models of Natural Language

- For example, Ferrer i Cancho & Solé (2003) propose that, for a source random variable *M*, natural languages *L* are minima of:

Information-Theoretic Models of Natural Language

- For example, Ferrer i Cancho & Solé (2003) propose that, for a source random variable *M*, natural languages *L* are minima of:

$$J_M(L) = H[M \mid L] + \lambda H[L]$$

Information-Theoretic Models of Natural Language

- For example, Ferrer i Cancho & Solé (2003) propose that, for a source random variable *M*, natural languages *L* are minima of:

$$J_M(L) = H[M \mid L] + \lambda H[L]$$

Ambiguity of meaning of the signal.
(Conditional entropy of meaning given signal)

Information-Theoretic Models of Natural Language

- For example, Ferrer i Cancho & Solé (2003) propose that, for a source random variable *M*, natural languages *L* are minima of:

$$J_M(L) = H[M \mid L] + \lambda H[L]$$

Ambiguity of meaning of the signal.
(Conditional entropy of meaning given signal)

Effort of using the signal
(Entropy of the signal)

Information-Theoretic Models of Natural Language

- For example, Ferrer i Cancho & Solé (2003) propose that, for a source random variable *M*, natural languages *L* are minima of:

$$J_M(L) = \underbrace{H[M \mid L]}_{\substack{\text{Ambiguity of meaning of the signal.} \\ \text{(Conditional entropy of meaning given signal)}}} + \lambda \underbrace{H[L]}_{\substack{\text{Effort of using the signal} \\ \text{(Entropy of the signal)}}}$$

- This function is also known as the **Deterministic Information Bottleneck** (Strouse & Schwab, 2016) and the **Infomax Criterion** (Bell & Sejnowski, 1995; Friston, 2010).

Information-Theoretic Models of Natural Language

- For example, Ferrer i Cancho & Solé (2003) propose that, for a source random variable *M*, natural languages *L* are minima of:

$$J_M(L) = H[M \mid L] + \lambda H[L]$$

Ambiguity of meaning of the signal.
(Conditional entropy of meaning given signal)

Effort of using the signal
(Entropy of the signal)

- This function is also known as the **Deterministic Information Bottleneck** (Strouse & Schwab, 2016) and the **Infomax Criterion** (Bell & Sejnowski, 1995; Friston, 2010).
- Key part: **effort is quantified using entropy (average surprisal).**

Efficiency Models of Word Order

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍
  - Bob threw the trash out. 👍

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍
  - Bob threw the trash out. 👍

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍
  - Bob threw the trash out. 👍
  - Bob threw out the old trash that had been sitting in the kitchen. 👍

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍
  - Bob threw the trash out. 👍
  - Bob threw out the old trash that had been sitting in the kitchen. 👍

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

  - Bob threw out the trash. 👍
  - Bob threw the trash out. 👍
  - Bob threw out the old trash that had been sitting in the kitchen. 👍
  - Bob threw the old trash that had been sitting in the kitchen out. 👎

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality** (aka dependency length minimization, dependency distance minimization, domain minimization, early immediate constituents, principle of head proximity, Behaghel's First Law, …)

- Bob threw out the trash. 👍
- Bob threw the trash out. 👍
- Bob threw out the old trash that had been sitting in the kitchen. 👍
- Bob threw the old trash that had been sitting in the kitchen out. 👎

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
    - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
    - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)
    - So the **linear distance** between words in dependencies **should be minimized** (for recent reviews, see Dyer, 2017; Temperley & Gildea, 2018; Liu et al., 2018).

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
  - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)
  - So the **linear distance** between words in dependencies **should be minimized** (for recent reviews, see Dyer, 2017; Temperley & Gildea, 2018; Liu et al., 2018).
  - Explains **pervasive word order patterns** across languages:

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
  - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)
  - So the **linear distance** between words in dependencies **should be minimized** (for recent reviews, see Dyer, 2017; Temperley & Gildea, 2018; Liu et al., 2018).
  - Explains **pervasive word order patterns** across languages:
    - Harmonic word order correlations (Greenberg, 1963; Hawkins, 1994)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
  - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)
  - So the **linear distance** between words in dependencies **should be minimized** (for recent reviews, see Dyer, 2017; Temperley & Gildea, 2018; Liu et al., 2018).
  - Explains **pervasive word order patterns** across languages:
    - Harmonic word order correlations (Greenberg, 1963; Hawkins, 1994)
    - Short-before-long and long-before-short preferences (Hawkins, 1994, 2004, 2014; Wasow, 2002)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
  - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)
  - So the **linear distance** between words in dependencies **should be minimized** (for recent reviews, see Dyer, 2017; Temperley & Gildea, 2018; Liu et al., 2018).
  - Explains **pervasive word order patterns** across languages:
    - Harmonic word order correlations (Greenberg, 1963; Hawkins, 1994)
    - Short-before-long and long-before-short preferences (Hawkins, 1994, 2004, 2014; Wasow, 2002)
    - Tendency to projectivity (Ferrer-i-Cancho, 2006)

Efficiency Models of Word Order

- The most powerful efficiency-based model of **word order** in natural language is **dependency locality:**
  - Robust evidence from psycholinguistics that **long dependencies cause processing difficulty** (Gibson, 1998, 2000; Grodner & Gibson, 2005; Bartek et al., 2011)
  - So the **linear distance** between words in dependencies **should be minimized** (for recent reviews, see Dyer, 2017; Temperley & Gildea, 2018; Liu et al., 2018).
  - Explains **pervasive word order patterns** across languages:
    - Harmonic word order correlations (Greenberg, 1963; Hawkins, 1994)
    - Short-before-long and long-before-short preferences (Hawkins, 1994, 2004, 2014; Wasow, 2002)
    - Tendency to projectivity (Ferrer-i-Cancho, 2006)

Focus of this Work

Focus of this Work

- **Problem**. Dependency locality is motivated in terms of heuristic arguments about memory usage.

Focus of this Work

- **Problem**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **Question**. How does dependency locality fit in formally with information-theoretic models of natural language?

Focus of this Work

- **Problem**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **Question**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **Answer**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:

Focus of this Work

- **<u>Problem</u>**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **<u>Question</u>**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **<u>Answer</u>**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:
    - **<u>Information locality</u>**: Words are under pressure to be close in proportion to their **mutual information.**

Focus of this Work

- **Problem**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **Question**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **Answer**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:
  - **Information locality**: Words are under pressure to be close in proportion to their **mutual information.**
- I show that **information locality makes correct predictions** beyond dependency locality in two domains:

Focus of this Work

- **Problem**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **Question**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **Answer**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:
  - **Information locality**: Words are under pressure to be close in proportion to their **mutual information.**
- I show that **information locality makes correct predictions** beyond dependency locality in two domains:
  - (1) Differences between different dependencies

Focus of this Work

- **Problem**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **Question**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **Answer**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:
  - **Information locality**: Words are under pressure to be close in proportion to their **mutual information.**
- I show that **information locality makes correct predictions** beyond dependency locality in two domains:
  - (1) Differences between different dependencies
  - (2) Relative order of adjectives

Focus of this Work

- **<u>Problem</u>**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **<u>Question</u>**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **<u>Answer</u>**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:
  - **<u>Information locality</u>**: Words are under pressure to be close in proportion to their **mutual information.**
- I show that **information locality makes correct predictions** beyond dependency locality in two domains:
  - (1) Differences between different dependencies
  - (2) Relative order of adjectives

Focus of this Work

- **<u>Problem</u>**. Dependency locality is motivated in terms of heuristic arguments about memory usage.
- **<u>Question</u>**. How does dependency locality fit in formally with information-theoretic models of natural language?
- **<u>Answer</u>**. When we adopt a **more sophisticated model of processing difficulty**, we can **derive dependency locality as a special case** of a new information-theoretic principle:
  - **<u>Information locality</u>**: Words are under pressure to be close in proportion to their **mutual information.**
- I show that **information locality makes correct predictions** beyond dependency locality in two domains:
  - (1) Differences between different dependencies
  - (2) Relative order of adjectives

Information Locality

- Introduction
- Information Locality
- Study 1: Strength of Dependencies
- Study 2: Adjective Order
- Conclusion

What makes words hard to process?

What makes words hard to process?

- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):

What makes words hard to process?

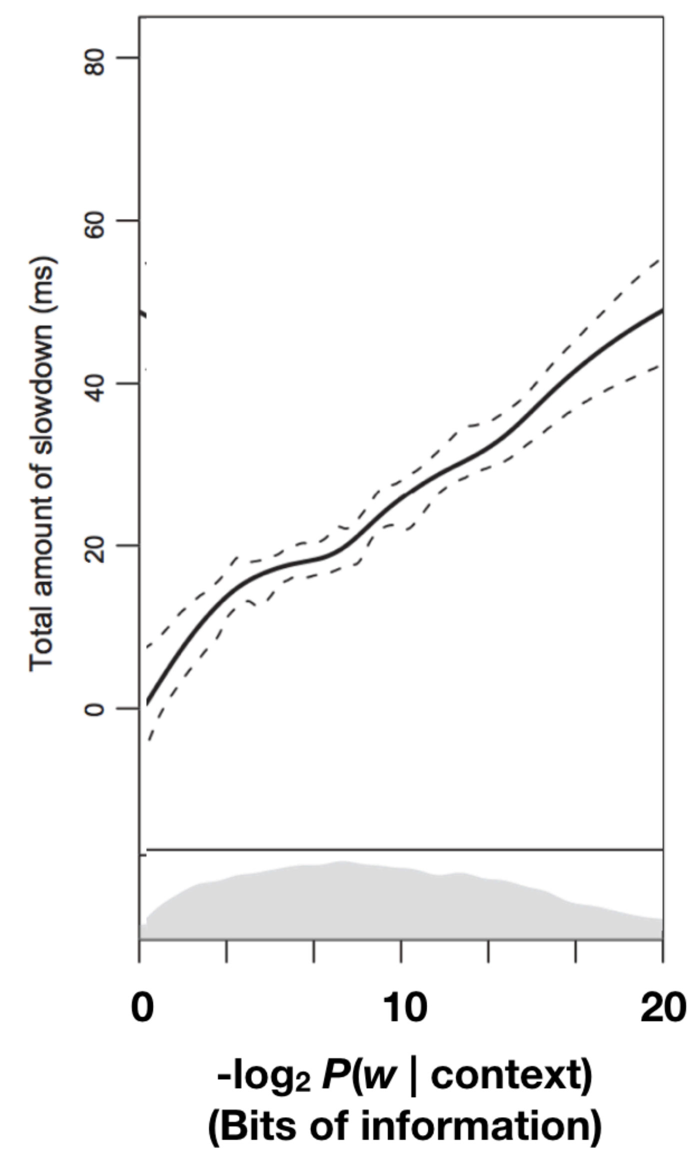
- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:

What makes words hard to process?

- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- *Difficulty*(w | context) = -log$P$(w | context)

What makes words hard to process?

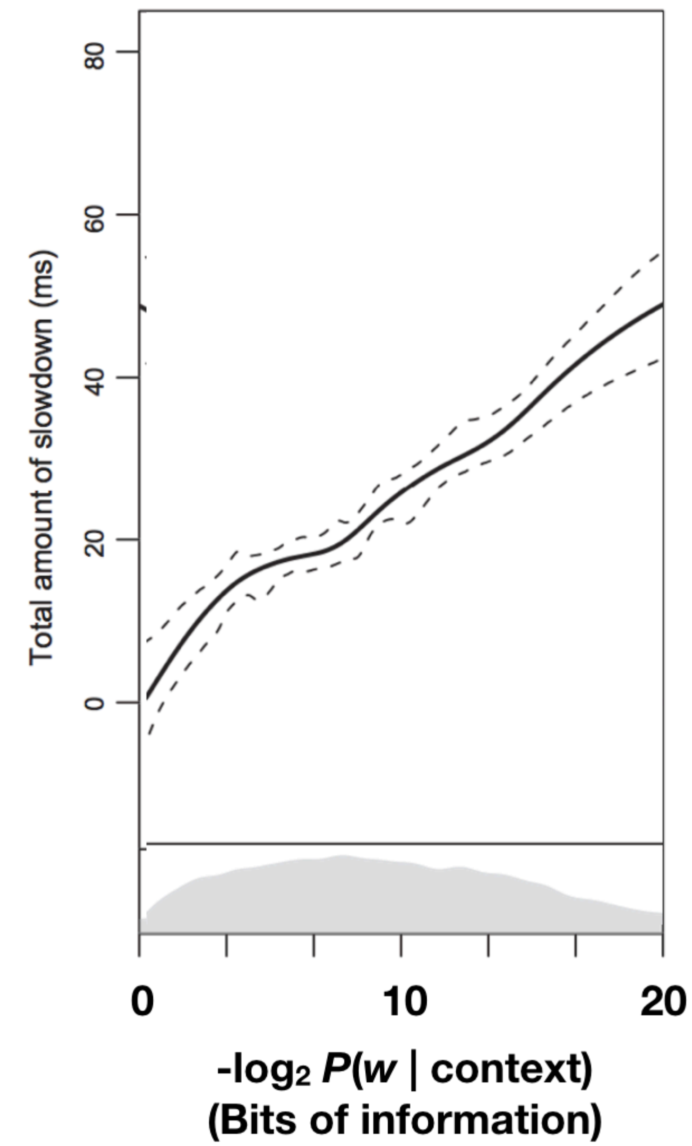
- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- *Difficulty*(*w* | context) = -log*P*(*w* | context)



Smith & Levy (2013). The effect of word predictability on reading time is logarithmic. *Cognition.*

What makes words hard to process?

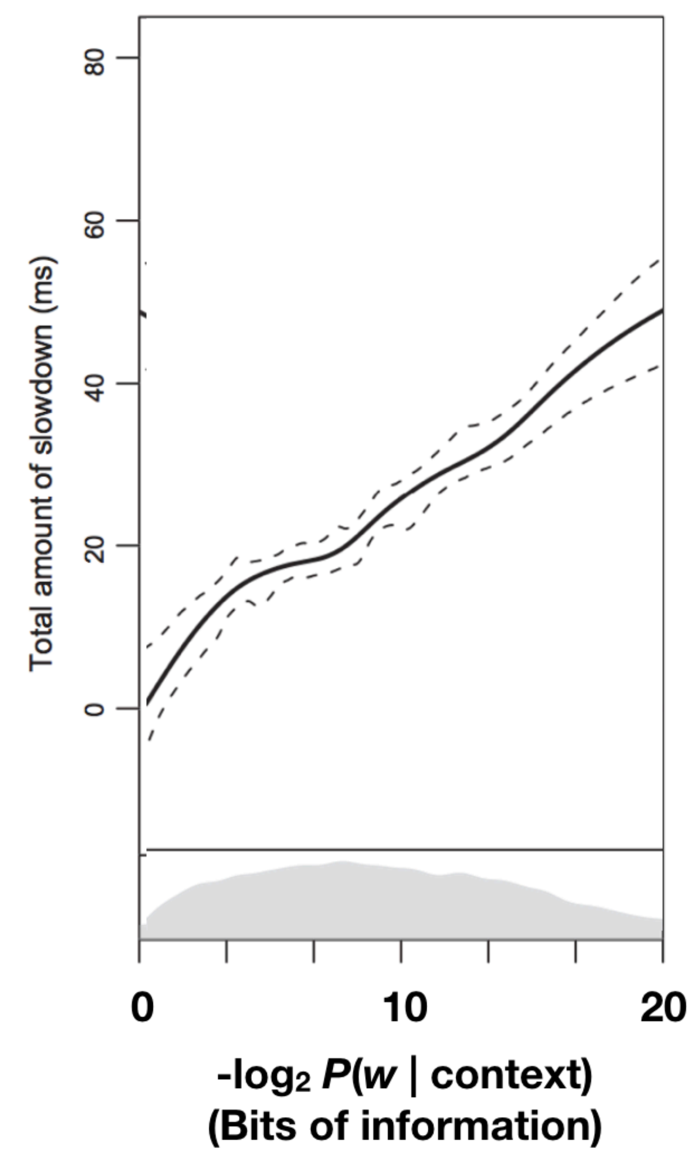
- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- $\mathit{Difficulty}(w \mid \text{context}) = -\log P(w \mid \text{context})$
- Accounts for:



Smith & Levy (2013). The effect of word predictability on reading time is logarithmic. *Cognition.*

What makes words hard to process?

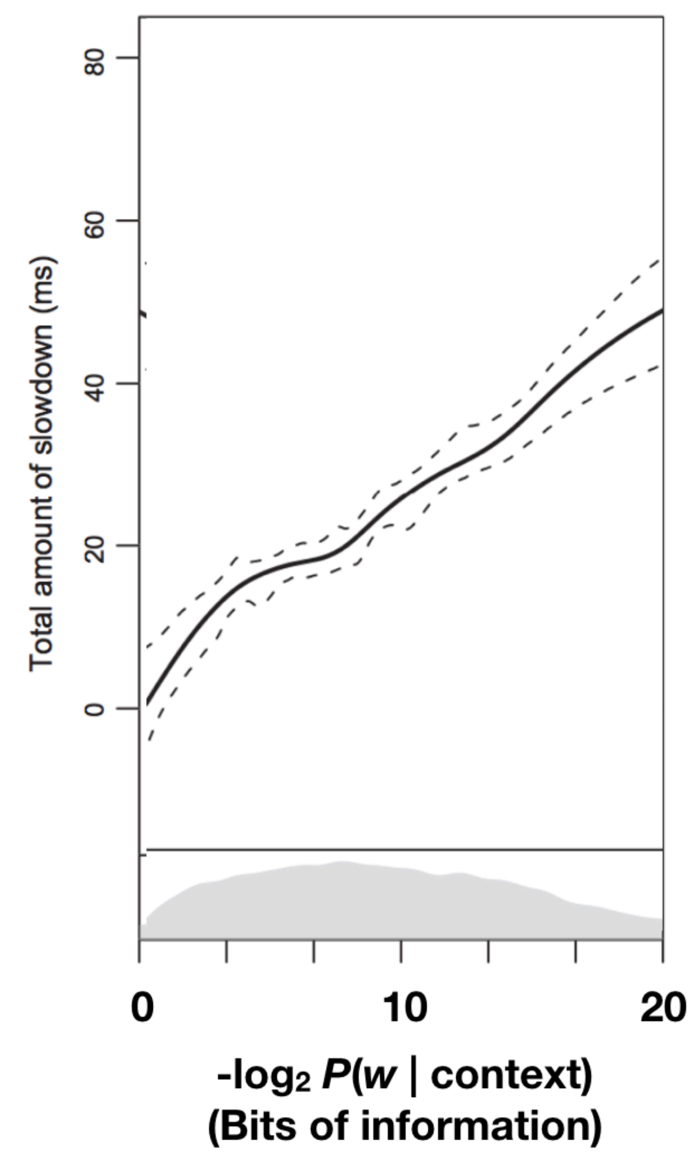
- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- *Difficulty*(w | context) = -log*P*(w | context)
- Accounts for:
    - **Garden path effects** (Hale, 2001)



Smith & Levy (2013). The effect of word predictability on reading time is logarithmic. *Cognition.*

What makes words hard to process?

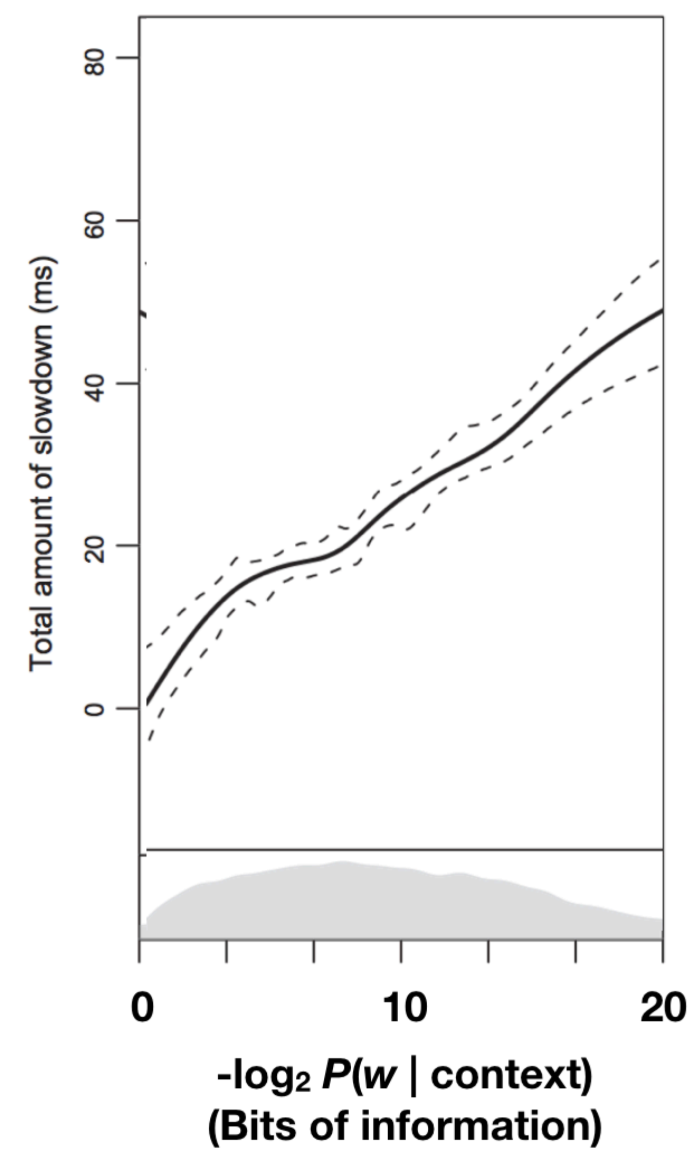
- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- $Difficulty(w \mid \text{context}) = -\log P(w \mid \text{context})$
- Accounts for:
  - **Garden path effects** (Hale, 2001)
  - **Antilocality effects** (Konieczny, 2000; Levy, 2008)



Smith & Levy (2013). The effect of word predictability on reading time is logarithmic. *Cognition.*

What makes words hard to process?

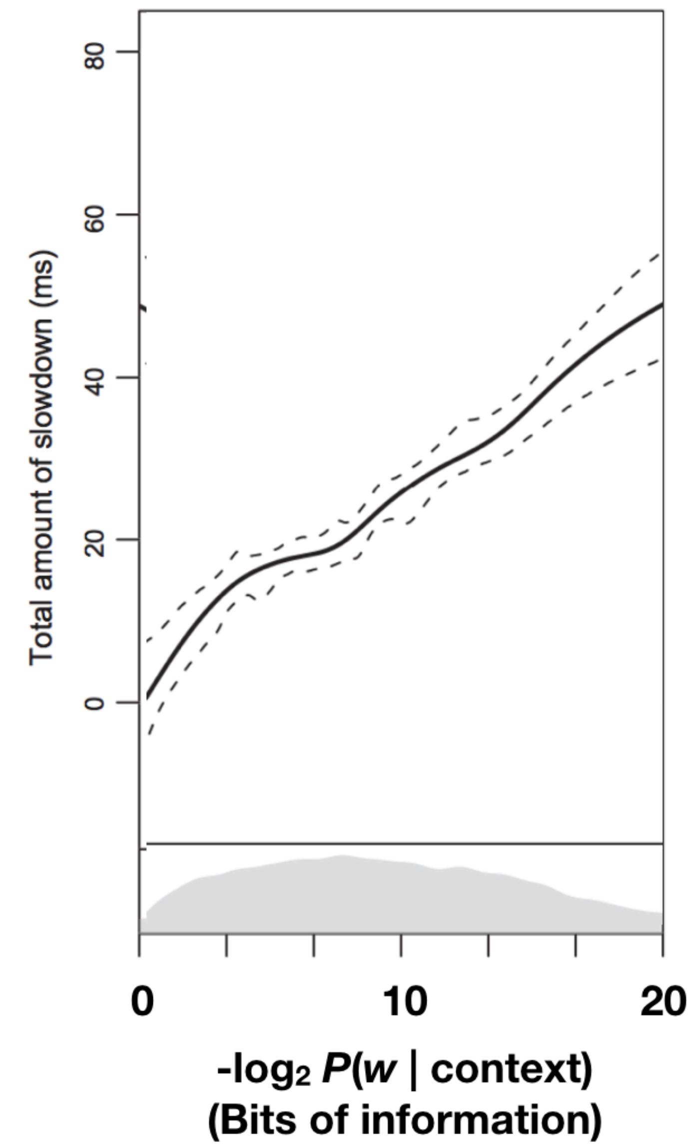
- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- *Difficulty*(*w* | context) = -log*P*(*w* | context)
- Accounts for:
  - **Garden path effects** (Hale, 2001)
  - **Antilocality effects** (Konieczny, 2000; Levy, 2008)
  - **Syntactic construction frequency effects** (Levy, 2008)



Smith & Levy (2013). The effect of word predictability on reading time is logarithmic. *Cognition.*

What makes words hard to process?

- **Surprisal theory** (Hale, 2001; Levy, 2008; Smith & Levy, 2013; Hale, 2016):
- **Processing difficulty at a word** is equal to **the surprisal of that word in context**:
- *Difficulty*(*w* | context) = -log*P*(*w* | context)
- Accounts for:
  - **Garden path effects** (Hale, 2001)
  - **Antilocality effects** (Konieczny, 2000; Levy, 2008)
  - **Syntactic construction frequency effects** (Levy, 2008)
- In other words, the **average processing difficulty** in a language is proportional to the **entropy of the language** $H[L]$.



Smith & Levy (2013). The effect of word predictability on reading time is logarithmic. *Cognition.*

Limitations of Surprisal Theory

Limitations of Surprisal Theory

Limitations of Surprisal Theory

Limitations of Surprisal Theory

Limitations of Surprisal Theory

- Surprisal theory has excellent empirical coverage for observable processing difficulty, *except*

Limitations of Surprisal Theory

- Surprisal theory has excellent empirical coverage for observable processing difficulty, *except*
- It **does not account for dependency locality effects** empirically (Levy, 2008, 2013) and provably cannot theoretically (Levy, 2006; Futrell, 2017).

Limitations of Surprisal Theory

- Surprisal theory has excellent empirical coverage for observable processing difficulty, *except*
- It **does not account for dependency locality effects** empirically (Levy, 2008, 2013) and provably cannot theoretically (Levy, 2006; Futrell, 2017).
- Reason: Surprisal theory has no notion of **memory limitations**.

Limitations of Surprisal Theory

- Surprisal theory has excellent empirical coverage for observable processing difficulty, *except*
- It **does not account for dependency locality effects** empirically (Levy, 2008, 2013) and provably cannot theoretically (Levy, 2006; Futrell, 2017).
- Reason: Surprisal theory has no notion of **memory limitations**.
- So how can we build memory limitations into surprisal theory?

Limitations of Surprisal Theory

- Surprisal theory has excellent empirical coverage for observable processing difficulty, *except*
- It **does not account for dependency locality effects** empirically (Levy, 2008, 2013) and provably cannot theoretically (Levy, 2006; Futrell, 2017).
- Reason: Surprisal theory has no notion of **memory limitations**.
- So how can we build memory limitations into surprisal theory?

Limitations of Surprisal Theory

- Surprisal theory has excellent empirical coverage for observable processing difficulty, *except*
- It **does not account for dependency locality effects** empirically (Levy, 2008, 2013) and provably cannot theoretically (Levy, 2006; Futrell, 2017).
- Reason: Surprisal theory has no notion of **memory limitations**.
- So how can we build memory limitations into surprisal theory?

How to fit memory into Surprisal Theory?

Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

- Surprisal: $\mathit{Diff}(w \mid \text{context}) = -\log P(w \mid \text{context})$

Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

- Surprisal: $Diff(w \mid \text{context}) = -\log P(w \mid \text{context})$

context	w
Bob threw the old trash that had been sitting in the kitchen	out

Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

- Surprisal: $Diff(w \mid \text{context}) = -\log P(w \mid \text{context})$

context	w
Bob threw the old trash that had been sitting in the kitchen	out

context

Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

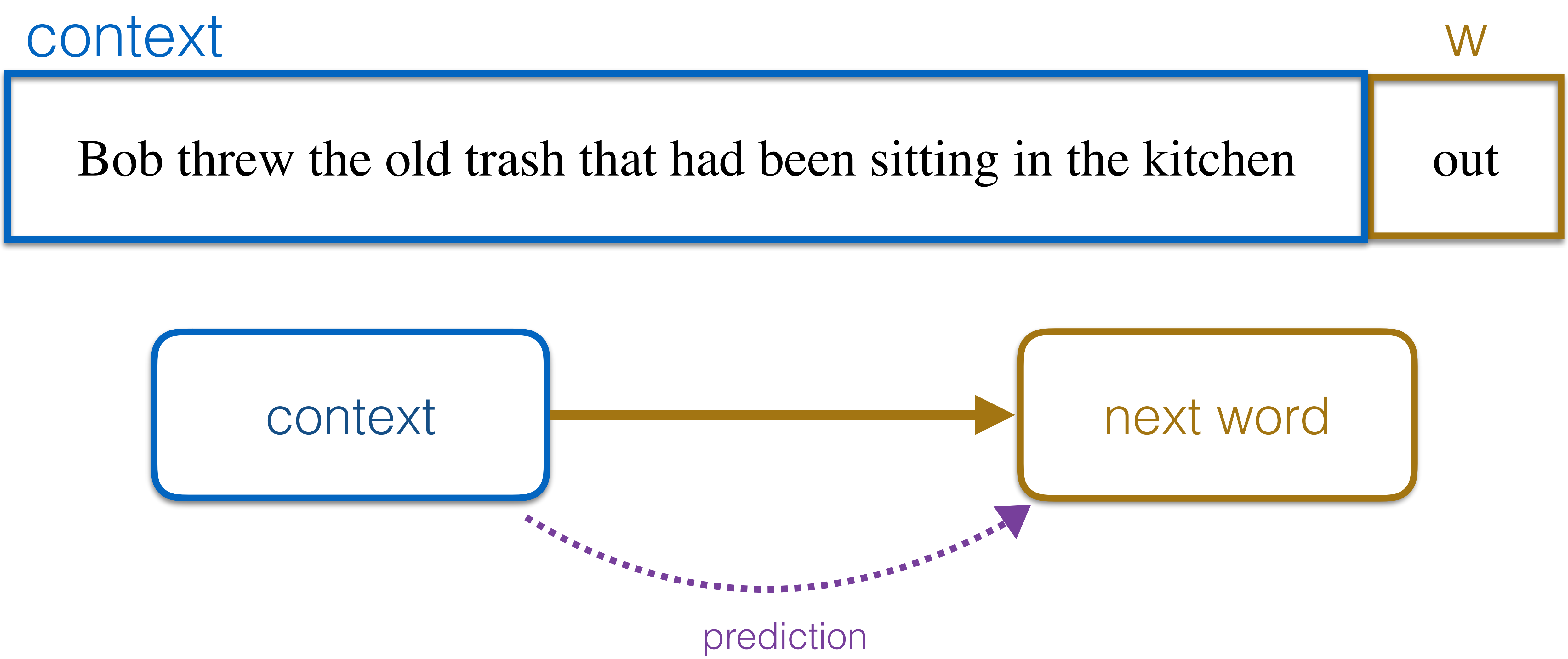
- Surprisal: $\mathit{Diff}(w \mid \text{context}) = -\log P(w \mid \text{context})$

context	w
Bob threw the old trash that had been sitting in the kitchen	out

Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

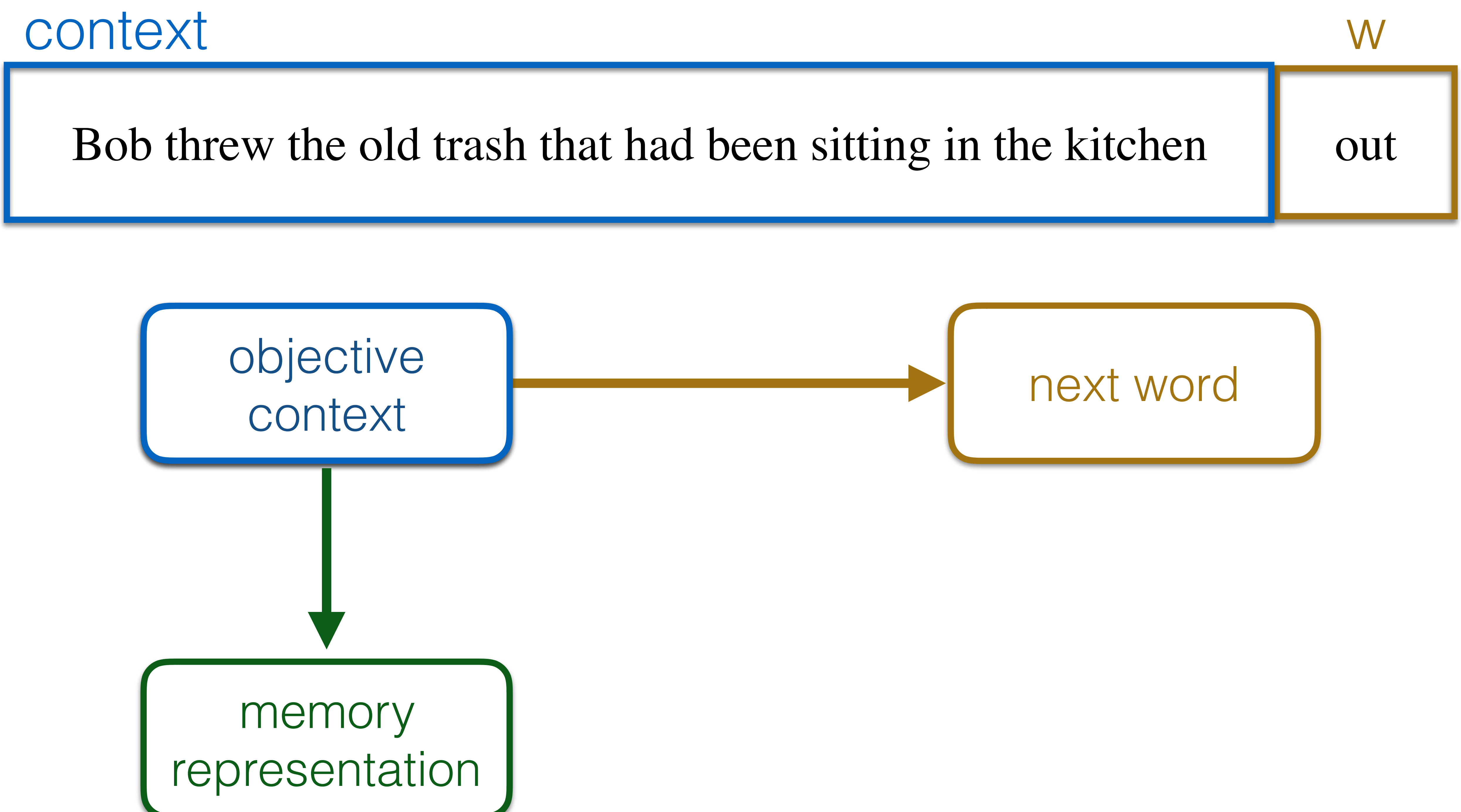
- Surprisal: $Diff(w \mid \text{context}) = -\log P(w \mid \text{context})$



Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

- Surprisal: $Diff(w \mid \text{context}) = -\log P(w \mid \text{context})$

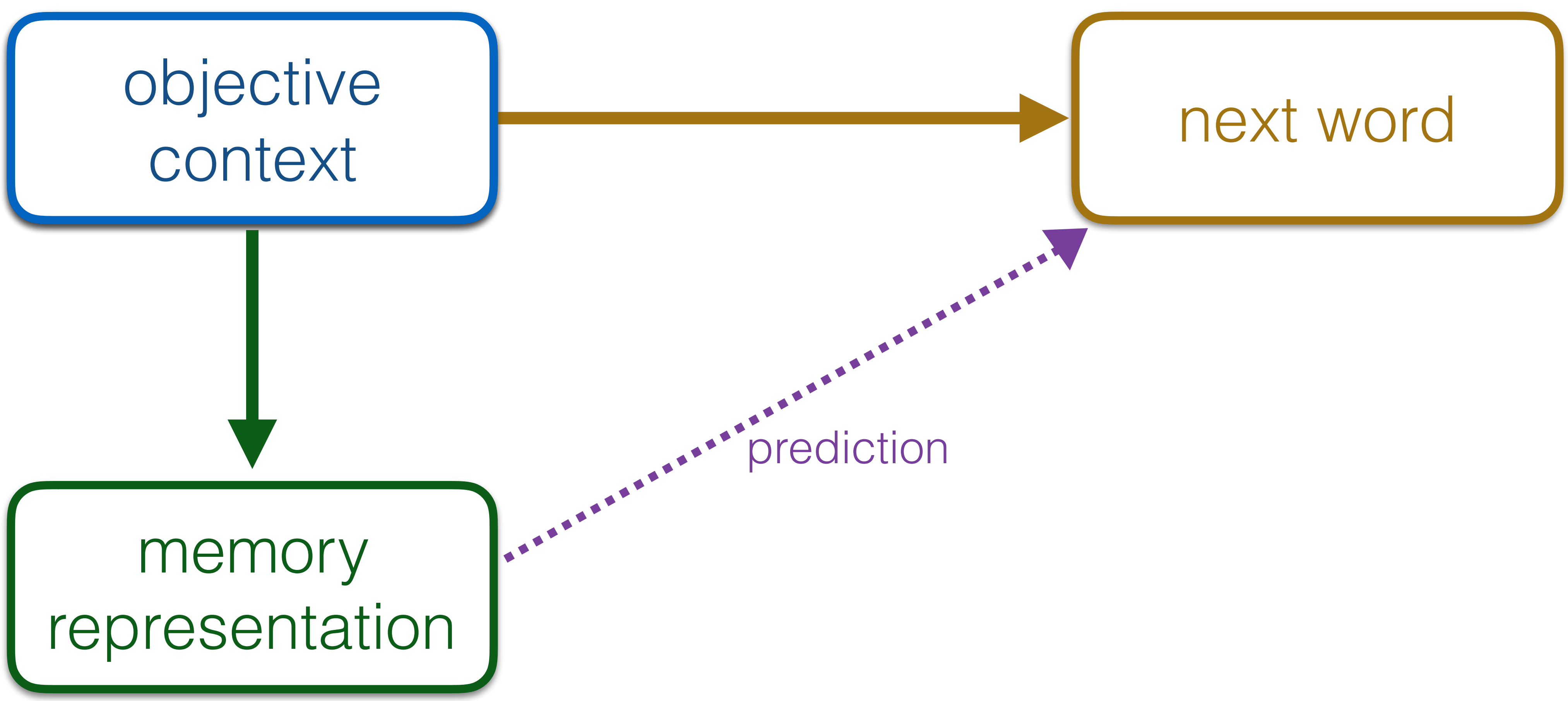


Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

- Surprisal: $Diff(w \mid \text{context}) = -\log P(w \mid \text{context})$

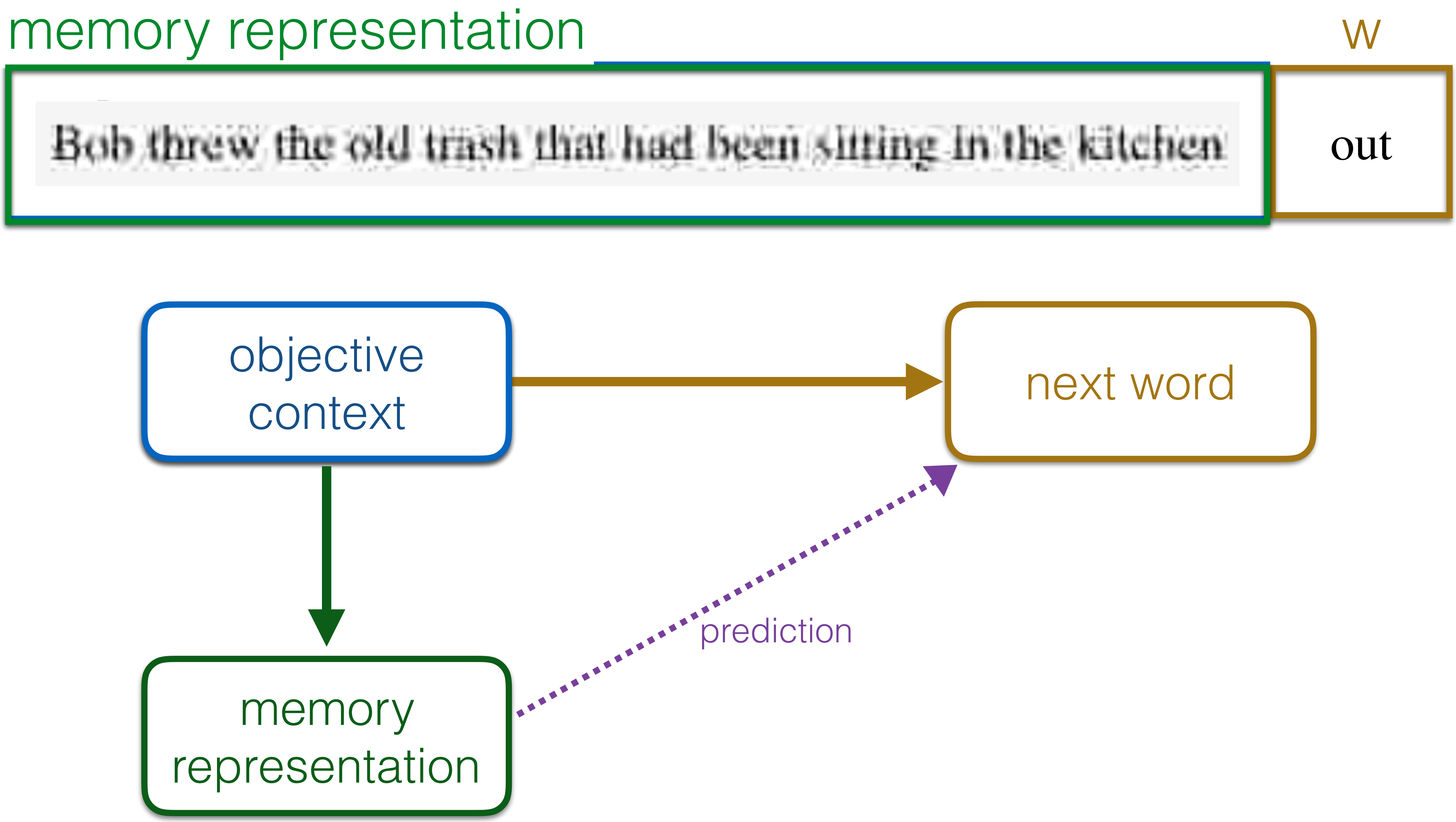
context	w
Bob threw the old trash that had been sitting in the kitchen	out



Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

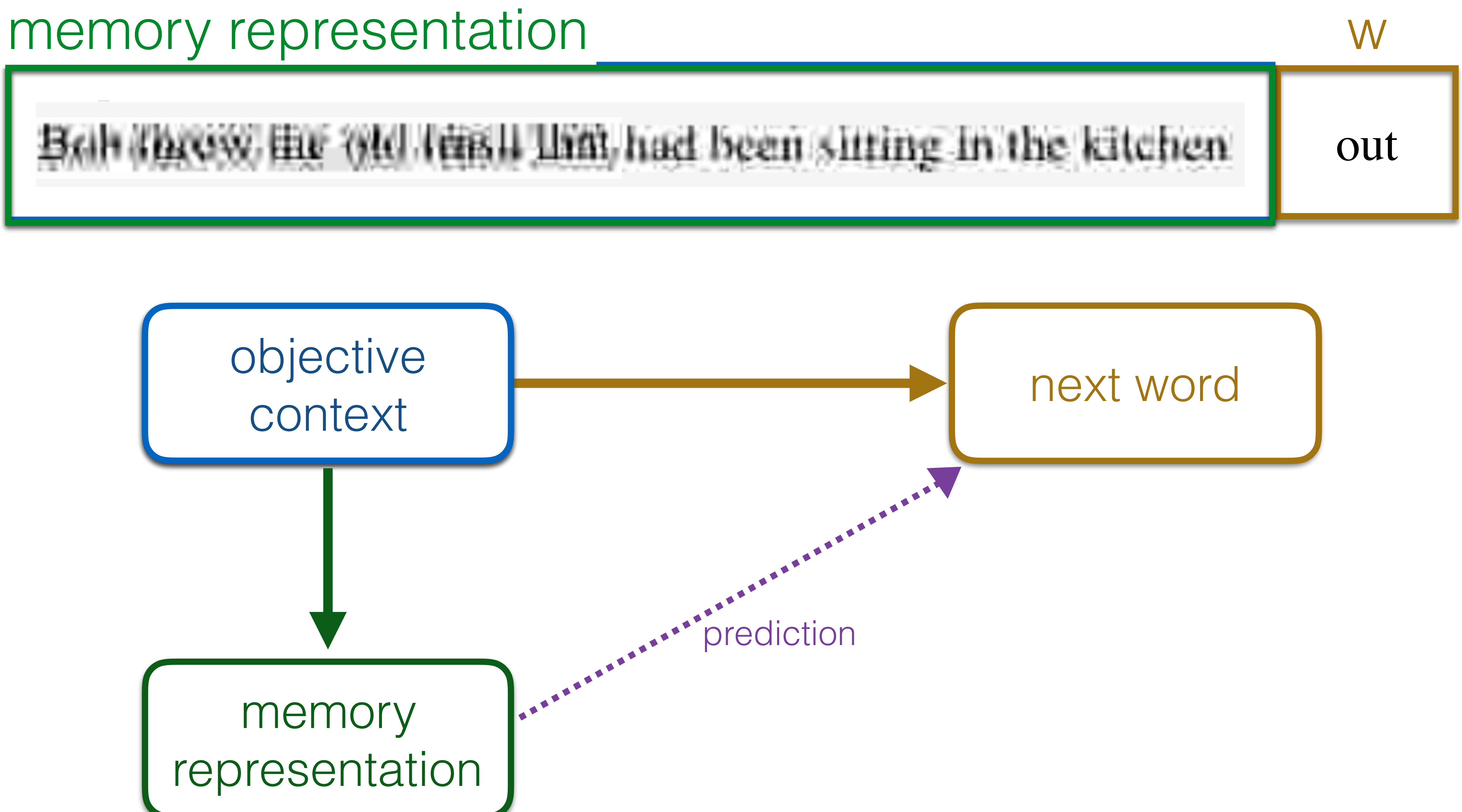
- Surprisal: $\mathit{Diff}(w \mid \text{context}) = -\log P(w \mid \text{context})$



Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

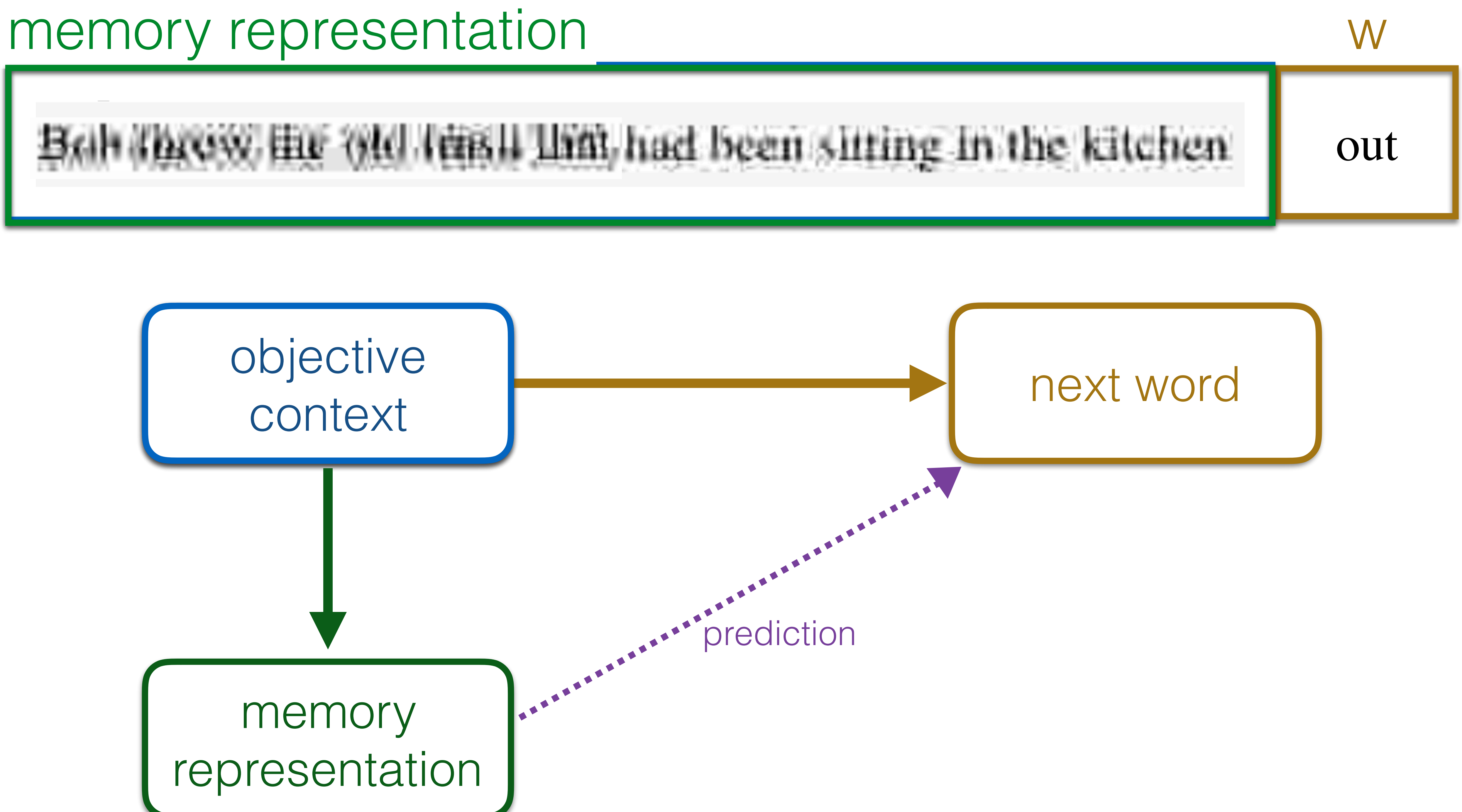
- Surprisal: $\mathit{Diff}(w \mid \text{context}) = -\log P(w \mid \text{context})$



Futrell & Levy (2017)

How to fit memory into Surprisal Theory?

- Lossy-context surprisal: $\mathit{Diff}(w \mid \text{context}) = -\log P(w \mid \text{memory representation})$



Futrell & Levy (2017)

What makes words hard to process?

What makes words hard to process?

- **Lossy-context surprisal**: Processing difficulty per word is

What makes words hard to process?

- **Lossy-context surprisal**: Processing difficulty per word is

$$\text{Diff}(w_i \mid w_{1,\ldots,i-1}) \propto -\log p(w_i \mid m_i),$$

What makes words hard to process?

- **Lossy-context surprisal**: Processing difficulty per word is

$$\mathrm{Diff}(w_i \mid w_{1,\ldots,i-1}) \propto -\log p(w_i \mid m_i),$$

where $\boldsymbol{m_i}$ **is a lossy compression of the context** $\boldsymbol{w_{1,\ldots,i-1}}$,
i.e. m_i is an approximate epsilon-machine (Feldman & Crutchfield, 1998; Marzen & Crutchfield, 2017).

What makes words hard to process?

- **Lossy-context surprisal**: Processing difficulty per word is

$$\mathrm{Diff}(w_i \mid w_{1,\ldots,i-1}) \propto -\log p(w_i \mid m_i),$$

where ***m_i* is a lossy compression of the context *$w_{1,\ldots,i-1}$***,
i.e. m_i is an approximate epsilon-machine (Feldman & Crutchfield, 1998; Marzen & Crutchfield, 2017).

- So the **average processing difficulty** for a language is a **cross entropy:**

What makes words hard to process?

- **Lossy-context surprisal**: Processing difficulty per word is

$$\text{Diff}(w_i \mid w_{1,\ldots,i-1}) \propto -\log p(w_i \mid m_i),$$

where ***m_i* is a lossy compression of the context *$w_{1,\ldots,i-1}$***,
i.e. m_i is an approximate epsilon-machine (Feldman & Crutchfield, 1998; Marzen & Crutchfield, 2017).

- So the **average processing difficulty** for a language is a **cross entropy:**

$$\text{Diff}(L) \propto \mathop{\mathbb{E}}_{w_{1,\ldots,i}} \left[-\log p(w_i \mid m_i)\right]$$

What makes words hard to process?

- **Lossy-context surprisal**: Processing difficulty per word is

$$\text{Diff}(w_i \mid w_{1,\ldots,i-1}) \propto -\log p(w_i \mid m_i),$$

where ***m_i* is a lossy compression of the context *$w_{1,\ldots,i-1}$***,
i.e. m_i is an approximate epsilon-machine (Feldman & Crutchfield, 1998; Marzen & Crutchfield, 2017).

- So the **average processing difficulty** for a language is a **cross entropy:**

$$\text{Diff}(L) \propto \mathop{\mathbb{E}}_{w_{1,\ldots,i}} [-\log p(w_i \mid m_i)]$$

$$\equiv H_L[L']$$

Information Locality

memory representation

Bob threw the old trash sitting in the kitchen	out

Information Locality

memory representation

Bob threew the old trash sitting in the kitchen | out

Information Locality

memory representation

Bob threew the old trash sitting in the kitchen | out

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.

Information Locality

memory representation

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.

Information Locality

memory representation

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.
- <u>Theorem</u> (Futrell & Levy, 2017):

Information Locality

memory representation

Bob threw the old trash sitting in the kitchen | out

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information are distant.**
- Theorem (Futrell & Levy, 2017):

$$\mathit{Diff}(w_i | w_1, \ldots, w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} e_{i-j} \operatorname{pmi}(w_i; w_j)$$

Information Locality

memory representation

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information are distant.**
- Theorem (Futrell & Levy, 2017):

$$\mathit{Diff}(w_i|w_1, \ldots, w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} e_{i-j} \boxed{\mathrm{pmi}(w_i; w_j)}$$

Information Locality

memory representation

Bob threw the old trash sitting in the kitchen	out

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.
- Theorem (Futrell & Levy, 2017):

$$\textit{Diff}(w_i|w_1, ..., w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} e_{i-j} \boxed{\mathrm{pmi}(w_i; w_j)}$$

Pointwise mutual information (pmi) is the most general statistical measure of *how strongly two values predict each other* (Church & Hanks, 1990)

$$\mathrm{pmi}(w; w') = \log \frac{p(w|w')}{p(w)}$$

Information Locality

memory representation

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.
- Theorem (Futrell & Levy, 2017):

$$\mathit{Diff}(w_i|w_1,\ldots,w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} e_{i-j} \boxed{\mathrm{pmi}(w_i;w_j)}$$

Information Locality

memory representation

Bob threw the old trash sitting in the kitchen | out

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.
- Theorem (Futrell & Levy, 2017):

$$\mathit{Diff}(w_i|w_1, \ldots, w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} e_{i-j}\,\mathrm{pmi}(w_i; w_j)$$

Information Locality

memory representation

Bob threw the old trash sitting in the kitchen | out

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.
- Theorem (Futrell & Levy, 2017):

$$\textit{Diff}(w_i|w_1, ..., w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} \boxed{e_{i-j}} \mathrm{pmi}(w_i; w_j)$$

Information Locality

memory representation

Bob threw the old trash sitting in the kitchen

out

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information** are distant.
- Theorem (Futrell & Levy, 2017):

$$\textit{Diff}(w_i|w_1, ..., w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} \boxed{e_{i-j}} \mathrm{pmi}(w_i; w_j)$$

e_d : Proportion of information retained about the *d*'th most recent word
(Under the noisy memory model, this must decrease monotonically.)

Information Locality

memory representation

- If information about words is lost at a constant rate (**noisy memory**), then the memory representation will have **less information about words that have been in memory longer**.
- This leads to **information locality**. **Difficulty increases when words with high mutual information are distant.**
- Theorem (Futrell & Levy, 2017):

$$\mathit{Diff}(w_i|w_1, ..., w_{i-1}) \approx -\log P(w) - \sum_{j=1}^{i-1} e_{i-j}\mathrm{pmi}(w_i; w_j)$$

e_d : Proportion of information retained about the *d*'th most recent word
(Under the noisy memory model, this must decrease monotonically.)

Information Locality

Information Locality

- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.

Information Locality

- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.
- How does this relate to **dependency locality**?

Information Locality

- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.
- How does this relate to **dependency locality**?
- Linking Hypothesis: **Words in syntactic dependencies have high mutual information** (de Paiva Alves, 1996; Yuret, 1998)

Information Locality

- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.
- How does this relate to **dependency locality**?
- Linking Hypothesis: **Words in syntactic dependencies have high mutual information** (de Paiva Alves, 1996; Yuret, 1998)
    - Makes sense a priori: Mutual information is a measure of **strength of covariance**.

Information Locality

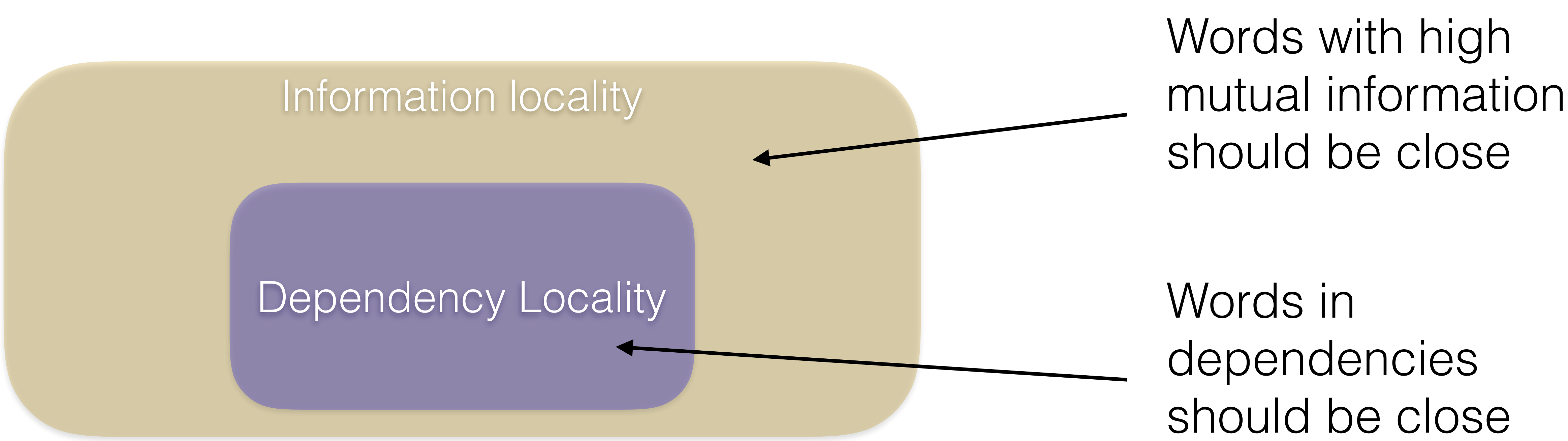
- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.
- How does this relate to **dependency locality**?
- Linking Hypothesis: **Words in syntactic dependencies have high mutual information** (de Paiva Alves, 1996; Yuret, 1998)
    - Makes sense a priori: Mutual information is a measure of **strength of covariance**.
    - If this is true, then we can see **dependency locality effects as a subset of information locality effects**.

Information Locality

- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.
- How does this relate to **dependency locality**?
- Linking Hypothesis: **Words in syntactic dependencies have high mutual information** (de Paiva Alves, 1996; Yuret, 1998)
    - Makes sense a priori: Mutual information is a measure of **strength of covariance**.
    - If this is true, then we can see **dependency locality effects as a subset of information locality effects**.
- I have a talk about this tomorrow! (Futrell, Qian, Gibson, Fedorenko & Blank, 2019).

Information Locality

- **Information locality:** I predict processing difficulty when words that predict each other (have high mutual information) are far apart.
- How does this relate to **dependency locality**?
- Linking Hypothesis: **Words in syntactic dependencies have high mutual information** (de Paiva Alves, 1996; Yuret, 1998)
  - Makes sense a priori: Mutual information is a measure of **strength of covariance**.
  - If this is true, then we can see **dependency locality effects as a subset of information locality effects**.
- I have a talk about this tomorrow! (Futrell, Qian, Gibson, Fedorenko & Blank, 2019).



Information Locality

- Introduction
- Information Locality
- Study 1: Strength of Dependencies
- Study 2: Adjective Order
- Conclusion

Strength of Dependencies

Strength of Dependencies

- **Dependency locality** says: All words in dependencies should be close.

Strength of Dependencies

- **Dependency locality** says: All words in dependencies should be close.
- **Information locality** says: Words want to be close in proportion to their mutual information.

Strength of Dependencies

- **Dependency locality** says: All words in dependencies should be close.
- **Information locality** says: Words want to be close in proportion to their mutual information.
- **Information locality prediction**: Words in dependencies **which predict each other other strongly** will be **especially attracted to each other**, beyond dependency locality effects.

Strength of Dependencies

- **Dependency locality** says: All words in dependencies should be close.
- **Information locality** says: Words want to be close in proportion to their mutual information.
- **Information locality prediction**: Words in dependencies **which predict each other other strongly** will be **especially attracted to each other**, beyond dependency locality effects.

Strength of Dependencies

Strength of Dependencies

- So: Fit a regression predicting the **distance between a head and dependent** from the **pmi of the head and dependent**.

Strength of Dependencies

- So: Fit a regression predicting the **distance between a head and dependent** from the **pmi of the head and dependent**.

$$y_r = \beta_0 + \beta_{\text{pmi}} \text{pmi}(h; d)$$

Strength of Dependencies

- So: Fit a regression predicting the **distance between a head and dependent** from the **pmi of the head and dependent**.

$$y_r = \beta_0 + \beta_{\text{pmi}} \text{pmi}(h; d)$$

Distance between words
in the *r'th* dependency in language *l*

Strength of Dependencies

- So: Fit a regression predicting the **distance between a head and dependent** from the **pmi of the head and dependent**.

$$y_r = \beta_0 + \beta_{\mathrm{pmi}} \mathrm{pmi}(h; d)$$

Distance between words
in the *r'th* dependency in language *l*

Strength of pmi-attraction effect

Strength of Dependencies

- So: Fit a regression predicting the **distance between a head and dependent** from the **pmi of the head and dependent**.

$$y_r = \beta_0 + \beta_{\text{pmi}} \text{pmi}(h; d)$$

Distance between words
in the *r'th* dependency in language *l*

Strength of pmi-attraction effect

- Fit to UD v2.1 corpora of 50 languages.

Strength of Dependencies

- So: Fit a regression predicting the **distance between a head and dependent** from the **pmi of the head and dependent**.

$$y_r = \beta_0 + \beta_{\text{pmi}}\text{pmi}(h; d)$$

Distance between words in the *r'th* dependency in language *l*

Strength of pmi-attraction effect

- Fit to UD v2.1 corpora of 50 languages.
- I measure pmi between POS tags, not wordforms, because wordform mutual information is hard to estimate for natural language (see my talk tomorrow)

Are words in dependencies with high pmi closer?

Are words in dependencies with high pmi closer?

- I find a significant pmi attraction effect in 48/50 languages.

Are words in dependencies with high pmi closer?

- I find a significant pmi attraction effect in 48/50 languages.

Language	β_{MI}	p
Ancient Greek	-0.18	<.001
Arabic	-0.26	<.001
Basque	-0.22	<.001
Belarusian	-0.20	<.001
Bulgarian	-0.29	<.001
Catalan	-0.29	<.001
Church Slavonic	-0.23	<.001
Coptic	-0.35	<.001
Croatian	-0.32	<.001
Czech	-0.27	<.001
Danish	-0.38	<.001
Dutch	-0.10	<.001
English	-0.38	<.001
Estonian	-0.32	<.001
Finnish	-0.29	<.001
French	-0.33	<.001
Galician	-0.35	<.001
German	-0.25	<.001
Gothic	-0.19	<.001
Hebrew	-0.21	<.001

Language	β_{MI}	p
Hindi	-0.26	<.001
Hungarian	-0.11	<.001
Indonesian	-0.22	<.001
Irish	-0.37	<.001
Italian	-0.35	<.001
Japanese	-0.32	<.001
Kazakh	-1.18	0.01
Korean	-0.14	<.001
Latin	-0.18	<.001
Latvian	-0.32	<.001
Lithuanian	-0.41	<.001
Mandarin	-0.19	<.001
Modern Greek	-0.25	<.001
Norwegian	-0.37	<.001
Persian	-0.19	<.001
Polish	-0.23	<.001
Portuguese	-0.23	<.001
Romanian	-0.36	<.001
Russian	-0.18	<.001
Sanskrit	0.10	0.28

Language	β_{MI}	p
Slovak	-0.30	<.001
Slovenian	-0.38	<.001
Spanish	-0.37	<.001
Swedish	-0.35	<.001
Tamil	-0.18	<.001
Turkish	-0.22	<.001
Ukrainian	-0.29	<.001
Urdu	-0.22	<.001
Uyghur	-0.04	0.79
Vietnamese	-0.27	<.001

Are words in dependencies with high pmi closer?

- I find a significant pmi attraction effect in 48/50 languages.
- Average effect size is -0.3:

Language	β_{MI}	p
Ancient Greek	-0.18	<.001
Arabic	-0.26	<.001
Basque	-0.22	<.001
Belarusian	-0.20	<.001
Bulgarian	-0.29	<.001
Catalan	-0.29	<.001
Church Slavonic	-0.23	<.001
Coptic	-0.35	<.001
Croatian	-0.32	<.001
Czech	-0.27	<.001
Danish	-0.38	<.001
Dutch	-0.10	<.001
English	-0.38	<.001
Estonian	-0.32	<.001
Finnish	-0.29	<.001
French	-0.33	<.001
Galician	-0.35	<.001
German	-0.25	<.001
Gothic	-0.19	<.001
Hebrew	-0.21	<.001

Language	β_{MI}	p
Hindi	-0.26	<.001
Hungarian	-0.11	<.001
Indonesian	-0.22	<.001
Irish	-0.37	<.001
Italian	-0.35	<.001
Japanese	-0.32	<.001
Kazakh	-1.18	0.01
Korean	-0.14	<.001
Latin	-0.18	<.001
Latvian	-0.32	<.001
Lithuanian	-0.41	<.001
Mandarin	-0.19	<.001
Modern Greek	-0.25	<.001
Norwegian	-0.37	<.001
Persian	-0.19	<.001
Polish	-0.23	<.001
Portuguese	-0.23	<.001
Romanian	-0.36	<.001
Russian	-0.18	<.001
Sanskrit	0.10	0.28

Language	β_{MI}	p
Slovak	-0.30	<.001
Slovenian	-0.38	<.001
Spanish	-0.37	<.001
Swedish	-0.35	<.001
Tamil	-0.18	<.001
Turkish	-0.22	<.001
Ukrainian	-0.29	<.001
Urdu	-0.22	<.001
Uyghur	-0.04	0.79
Vietnamese	-0.27	<.001

Are words in dependencies with high pmi closer?

- I find a significant pmi attraction effect in 48/50 languages.
- Average effect size is -0.3:
  - For each bit of pmi between two words, they are 0.3 words closer together on average.

Language	β_{MI}	p
Ancient Greek	-0.18	<.001
Arabic	-0.26	<.001
Basque	-0.22	<.001
Belarusian	-0.20	<.001
Bulgarian	-0.29	<.001
Catalan	-0.29	<.001
Church Slavonic	-0.23	<.001
Coptic	-0.35	<.001
Croatian	-0.32	<.001
Czech	-0.27	<.001
Danish	-0.38	<.001
Dutch	-0.10	<.001
English	-0.38	<.001
Estonian	-0.32	<.001
Finnish	-0.29	<.001
French	-0.33	<.001
Galician	-0.35	<.001
German	-0.25	<.001
Gothic	-0.19	<.001
Hebrew	-0.21	<.001
Hindi	-0.26	<.001
Hungarian	-0.11	<.001
Indonesian	-0.22	<.001
Irish	-0.37	<.001
Italian	-0.35	<.001
Japanese	-0.32	<.001
Kazakh	-1.18	0.01
Korean	-0.14	<.001
Latin	-0.18	<.001
Latvian	-0.32	<.001
Lithuanian	-0.41	<.001
Mandarin	-0.19	<.001
Modern Greek	-0.25	<.001
Norwegian	-0.37	<.001
Persian	-0.19	<.001
Polish	-0.23	<.001
Portuguese	-0.23	<.001
Romanian	-0.36	<.001
Russian	-0.18	<.001
Sanskrit	0.10	0.28
Slovak	-0.30	<.001
Slovenian	-0.38	<.001
Spanish	-0.37	<.001
Swedish	-0.35	<.001
Tamil	-0.18	<.001
Turkish	-0.22	<.001
Ukrainian	-0.29	<.001
Urdu	-0.22	<.001
Uyghur	-0.04	0.79
Vietnamese	-0.27	<.001

Information Locality

- Introduction
- Information Locality
- Study 1: Strength of Dependencies
- Study 2: Adjective Order
- Conclusion

Adjective Order Constraints

Adjective Order Constraints

The pretty red Italian car 👍

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

- There are **constraints on relative order of adjectives** that are stable across speakers and languages.

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

- There are **constraints on relative order of adjectives** that are stable across speakers and languages.
- Strongest empirical generalization: more *subjective* adjectives are farther out (Scontras et al., 2017)

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

- There are **constraints on relative order of adjectives** that are stable across speakers and languages.
- Strongest empirical generalization: more *subjective* adjectives are farther out (Scontras et al., 2017)
- <u>Information locality explanation</u>: **Adjectives with high pmi with a noun will appear relatively close to that noun.**

Adjective Order Constraints

The pretty red Italian car 👍

The red pretty Italian car 👎

The Italian pretty red car 👎

The pretty Italian red car 👎

- There are **constraints on relative order of adjectives** that are stable across speakers and languages.
- Strongest empirical generalization: more *subjective* adjectives are farther out (Scontras et al., 2017)
- Information locality explanation: **Adjectives with high pmi with a noun will appear relatively close to that noun.**
  - Possibly conceptually related to subjectivity.

Does Adjective Order Correspond to Mutual Information?

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?
- Data: Google Syntactic n-Grams (8.5 billion adjective-noun pairs)

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?
- Data: Google Syntactic n-Grams (8.5 billion adjective-noun pairs)
- Model: Logistic regression predicting order from pmi.

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?
- Data: Google Syntactic n-Grams (8.5 billion adjective-noun pairs)
- Model: Logistic regression predicting order from pmi.
- **Result: PMI predicts adjective order** for held-out data with **66.9%** accuracy.

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?
- Data: Google Syntactic n-Grams (8.5 billion adjective-noun pairs)
- Model: Logistic regression predicting order from pmi.
- **Result: PMI predicts adjective order** for held-out data with **66.9%** accuracy.
  - Best previously known predictor (**subjectivity**) gets **68.4%**

Does Adjective Order Correspond to Mutual Information?

- 1. Gather a large set of adjective—adjective—noun triples from a corpus.
- 2. Measure pmi between adjectives and nouns.
- 3. Does pmi(A;N) predict that A will be closer to the noun than the other adjective?
- Data: Google Syntactic n-Grams (8.5 billion adjective-noun pairs)
- Model: Logistic regression predicting order from pmi.
- **Result: PMI predicts adjective order** for held-out data with **66.9%** accuracy.
  - Best previously known predictor (**subjectivity**) gets **68.4%**
  - **PMI + Subjectivity** gets **72.9%** accuracy

Discussion

Discussion

- Other theories aim to explain the same data…

Discussion

- Other theories aim to explain the same data…
    - Dyer's (2017, 2018) **Integration Cost**: Involves the conditional entropy of dependency relation labels given words.

Discussion

- Other theories aim to explain the same data…
  - Dyer’s (2017, 2018) **Integration Cost**: Involves the conditional entropy of dependency relation labels given words.
  - Hahn et al.’s (2018) **Subjective Rational Speech Acts Model**: Involves noisy incremental memory in the computation of meaning.

Discussion

- Other theories aim to explain the same data…
  - Dyer's (2017, 2018) **Integration Cost**: Involves the conditional entropy of dependency relation labels given words.
  - Hahn et al.'s (2018) **Subjective Rational Speech Acts Model**: Involves noisy incremental memory in the computation of meaning.
  - Scontras et al.'s (2019) **Noisy composition model** explains adjective order in terms of noisy hierarchical computation of meaning.

Discussion

- Other theories aim to explain the same data…
  - Dyer's (2017, 2018) **Integration Cost**: Involves the conditional entropy of dependency relation labels given words.
  - Hahn et al.'s (2018) **Subjective Rational Speech Acts Model**: Involves noisy incremental memory in the computation of meaning.
  - Scontras et al.'s (2019) **Noisy composition model** explains adjective order in terms of noisy hierarchical computation of meaning.
- Future work will have to rigorously disentangle the predictions of these theories.

Discussion

- Other theories aim to explain the same data…
  - Dyer's (2017, 2018) **Integration Cost**: Involves the conditional entropy of dependency relation labels given words.
  - Hahn et al.'s (2018) **Subjective Rational Speech Acts Model**: Involves noisy incremental memory in the computation of meaning.
  - Scontras et al.'s (2019) **Noisy composition model** explains adjective order in terms of noisy hierarchical computation of meaning.
- Future work will have to rigorously disentangle the predictions of these theories.
- Problem: The relevant information-theoretic quantities are hard to estimate accurately.

Information Locality

- Introduction
- Information Locality
- Study 1: Strength of Dependencies
- Study 2: Adjective Order
- Conclusion

Conclusion

Conclusion

- **<u>Question</u>. How does dependency locality fit in formally with information-theoretic models of natural language?**

Conclusion

- **<u>Question</u>. How does dependency locality fit in formally with information-theoretic models of natural language?**

$$J_M(L) = H[M \mid L] + \lambda H[L]$$

Conclusion

- **<u>Question</u>. How does dependency locality fit in formally with information-theoretic models of natural language?**

$$J_M(L) = H[M \mid L] + \lambda H_L[L']$$

Conclusion

- **<u>Question</u>. How does dependency locality fit in formally with information-theoretic models of natural language?**

$$J_M(L) = H[M \mid L] + \lambda H_L[L']$$

Dependency locality happens in this term in the form of information locality

Outstanding Questions for Information Locality

Outstanding Questions for Information Locality

- Does information locality capture the **trade-off** of **complex morphology** and **deterministic word order**? (Koplenig et al., 2017)

Outstanding Questions for Information Locality

- Does information locality capture the **trade-off** of **complex morphology** and **deterministic word order**? (Koplenig et al., 2017)
  - Depends on the precise relationship between morphology and inter-word MI.

Outstanding Questions for Information Locality

- Does information locality capture the **trade-off** of **complex morphology** and **deterministic word order**? (Koplenig et al., 2017)
  - Depends on the precise relationship between morphology and inter-word MI.
- Is the right notion of mutual information purely MI between words, or is it also something that **takes into account meaning**?

Outstanding Questions for Information Locality

- Does information locality capture the **trade-off** of **complex morphology** and **deterministic word order**? (Koplenig et al., 2017)
    - Depends on the precise relationship between morphology and inter-word MI.
- Is the right notion of mutual information purely MI between words, or is it also something that **takes into account meaning**?
    - E.g., dependency relation types, as in Dyer’s Integration Cost theory

Outstanding Questions for Information Locality

- Does information locality capture the **trade-off** of **complex morphology** and **deterministic word order**? (Koplenig et al., 2017)
    - Depends on the precise relationship between morphology and inter-word MI.
- Is the right notion of mutual information purely MI between words, or is it also something that **takes into account meaning**?
    - E.g., dependency relation types, as in Dyer's Integration Cost theory
- Does information locality make different predictions from dependency locality wrt **crossing dependencies**?

Outstanding Questions for Information Locality

- Does information locality capture the **trade-off** of **complex morphology** and **deterministic word order**? (Koplenig et al., 2017)
    - Depends on the precise relationship between morphology and inter-word MI.
- Is the right notion of mutual information purely MI between words, or is it also something that **takes into account meaning**?
    - E.g., dependency relation types, as in Dyer's Integration Cost theory
- Does information locality make different predictions from dependency locality wrt **crossing dependencies**?

Thanks all!

- All code is available online at http://github.com/langprocgroup/adjorder and http://github.com/langprocgroup/cliqs
- Thanks to Roger Levy, Ted Gibson, and Tim O'Donnell for discussions.
- Thanks to the SyntaxFest reviewers for helpful comments.
- Thanks to the Quasy organizers for a great conference!